

AFRL-IF-RS-TR-2005-255
Final Technical Report
July 2005



RESOURCE CONTROL IN LARGE-SCALE MOBILE-AGENTS SYSTEMS

Dartmouth College

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. J390

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2005-255 has been reviewed and is approved for publication

APPROVED: /s/

JOHN F. LEMMER
Project Engineer

FOR THE DIRECTOR: /s/

JAMES W. CUSACK, Chief
Information Systems Division
Information Directorate

REPORT DOCUMENTATION PAGE			<i>Form Approved</i> <i>OMB No. 074-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE JULY 2005	3. REPORT TYPE AND DATES COVERED Final Apr 98 – Dec 03	
4. TITLE AND SUBTITLE RESOURCE CONTROL IN LARGE-SCALE MOBILE-AGENTS SYSTEMS			5. FUNDING NUMBERS C - F30602-98-2-0107 PE - 637601E PR - AGEN TA - T0 WU - 03	
6. AUTHOR(S) David Kotz, Daniela Rus, George Cybenko and Robert S. Gray				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Dartmouth College Hanover New Hampshire 03755			8. PERFORMING ORGANIZATION REPORT NUMBER N/A	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Defense Advanced Research Projects Agency AFRL/IFSA 3701 North Fairfax Drive 525 Brooks Road Arlington Virginia 22203-1714 Rome New York 13441-4505			10. SPONSORING / MONITORING AGENCY REPORT NUMBER AFRL-IF-RS-TR-2005-255	
11. SUPPLEMENTARY NOTES AFRL Project Engineer: John F. Lemmer/IFSA/(315) 330-3657/ John.Lemmer@rl.af.mil				
12a. DISTRIBUTION / AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.				12b. DISTRIBUTION CODE
13. ABSTRACT (Maximum 200 Words) This project aimed at gaining a better understanding of mobile-agent systems, and in particular their scalability and ability to interoperate with other mobile-agent systems. Using models, simulation, and large-scale experiments, and collaborating with the developers of other mobile-agent systems, we performed the first and (to date) most extensive scalability analysis of mobile-agent software. We compared their performance on information-retrieval tasks common to many application domains. We explored their performance benefits in a wireless-network environment. We developed technology that allows a mobile agent written for one mobile-agent system to migrate to and execute in a different mobile-agent system. We also expanded our prior work on the security of mobile agents.				
14. SUBJECT TERMS Mobile-Agent Systems, Sensor Networks, Information Retrieval, Wireless Networks			15. NUMBER OF PAGES 43	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

Contents

1	Introduction	1
2	Personnel	1
2.1	Faculty	1
2.2	Research Staff	1
2.3	Administrative Staff	2
2.4	Post-doctoral fellows	2
2.5	Graduate students	2
2.6	Undergraduate students	3
3	Research products	3
3.1	Papers published	3
3.1.1	Mobile agents.	4
3.1.2	Market-based resource control.	5
3.1.3	Mobile computing.	5
3.1.4	Wireless networks: Using mobile agents for routing.	6
3.1.5	Mobile agents for information retrieval.	7
3.1.6	Other papers.	7
3.2	Software	8
4	Demonstrations and Presentations	9
4.1	Demonstrations	9
4.2	Presentations	9
5	Technology Transfer	13
6	Community service and advisory functions	13
7	Conclusions and lessons learned	16
	References	17

1 Introduction

Our group at Dartmouth College, known as the *D'Agents* project, was honored to be part of DARPA's Control of Agent-Based Systems (CoABS) project. We used this funding to delve ever more deeply into the fundamental challenges facing mobile agent systems, including related topics in mobile ad hoc wireless networks, sensor networks, market-based control mechanisms for mobile agent systems, information retrieval (a common application domain for mobile agents), and middleware for sensor information processing.

We collaborated closely with other research teams from Lockheed-Martin, the University of Western Florida, and Boeing, as well as non-CoABS researchers from the University of Illinois at Urbana-Champaign. Indeed, we led the Mobility TIE, a deep and productive collaboration that led to specifications for mobility support that became part of the CoABS Grid, and to two lines of research papers: one based on joint scalability experiments, and one based on joint development of an interoperability layer. The Mobility TIE also was involved in a Fleet Battle Experiment (obtaining experimental data that was incorporated into one of our papers [KCG⁺02]) and the CoAX coalition-forces demonstration.

The result was an extremely productive research team, producing over 80 papers, training over two dozen students, incorporating our research in undergraduate and graduate courses, demonstrating working prototypes to key players in government and industry, and laying the intellectual groundwork for several exciting new research efforts now underway.

In this final report, we document the people involved, the publications produced, the demonstrations completed, our efforts at technology transfer, our support of the broad research community, and our assistance to government agencies.

We summarize our publications in the Papers section below and detail them all in the References section. The papers are available at <http://agent.cs.dartmouth.edu/CoABS/>.

2 Personnel

All of the personnel listed here are (or were, at the time of their involvement in this project) at Dartmouth College. Not all of them were involved at the same time; some were involved for a few months or a few years of the project.

2.1 Faculty

We list each at their rank as of July 2003.

Javed Aslam, Assistant Professor, Department of Computer Science

George Cybenko, Professor, Thayer School of Engineering

David Kotz, Professor, Department of Computer Science

Daniela Rus, Professor, Department of Computer Science

2.2 Research Staff

Robert Gray, Ph.D., Research Associate, Thayer School of Engineering

Arne Grimstrup, Research Associate, Department of Computer Science

Dennis McGrath, Research Associate, Thayer School of Engineering

Ronald Peterson, Senior Programmer, Department of Computer Science

2.3 Administrative Staff

Catherine La Touche, administrative assistant, Department of Computer Science

Alison Sartono, administrative assistant, Department of Computer Science

2.4 Post-doctoral fellows

Zack Butler, Ph.D, Postdoctoral Research Associate, now an Assistant Professor of Computer Science at Rochester Institute of Technology

Guofei Jiang, Ph.D, Postdoctoral Research Associate, now at NEC Laboratories America, Inc.

2.5 Graduate students

Daniel Bilar, PhD 2003, after switching to a different project: “Quantitative Risk Analysis of Computer Networks”; now a visiting professor of computer science at Colby College in Maine.

Jonathan Bredin, Ph.D completed 2001: “Market-based Control of Mobile-agent Systems” [Bre01a]; now an Assistant Professor of Computer Science at Colorado College.

Guanling Chen, Ph.D completed August 2004: “Solar: Building A Context Fusion Network for Pervasive Computing” [Che04a]; will be Assistant Professor of Computer Science at the University of Massachusetts, Lowell.

Diego Hernando, M.S. completed 2004: “Entropy Rates and Sampling Theory for Hidden Markov Models”; entering University of Illinois-Urbana Champaign, this Fall in the PhD program.

Han Li, M.S. completed 2003: “Semantic Message Oriented Middleware”; now on the staff at the Thayer School of Engineering.

Qun Li, Ph.D completed 2004: “Mobility and Communication in Sensor Networks” [Li04]; will be Assistant Professor of Computer Science at William and Mary College.

Kazuhiro Minami, Ph.D expected summer 2005.

Katsuhiro Moizumi, Ph.D 1999: “Mobile Agent Planning” [Moi98]; now at Furukawa Electric Company in Japan.

Soumendra Nanda, Ph.D student currently.

Sidharth Nazareth, M.S. completed spring 2003, after switching to a different group: “SPADE: SPKI/SDSI for Attribute Release Policies in a Distributed Environment”; current location unknown.

Glenn Nofsinger, Ph.D expected summer 2005.

Yong Sheng, Ph.D expected Fall 2004.

Veeravanallur (Shankar) Sundaram, M.S. completed 2000: “Mission-flow Constructor: a workflow management system using mobile agents” [Sun00].

Ron Xie, left without graduating.

2.6 Undergraduate students

Numerous undergraduate students were involved in this project, whether for pay or for academic credit. Involved for as little as three months or as much as three years, they served as programmers, system administrators, application developers, or honors research students. Some even co-authored papers published in the literature. Several women and minorities were involved. A partial list of students includes Ryan Acton '00, Bonnie Archampong (research intern, summer 2002), Mihai Banulescu '99 (applications), Christian Bennett '99 (lab manager), Mike Brewer '99, James Chalfant '99 (agent-tracking tool), Ezra Cooper '00 (market-based control), Michael Corr (agent-tracking tool), Sergey Demidenko (research intern; a student at Clarkson University), Nikita Dubrovsky '04 (simulation), Joe Edelman '99 (electronic currency, yellow pages, applications), Aaron Fiske (ad hoc routing), Jennifer Grey (research intern, summer 2002), Adrian Michael Hartline '03, Alexander Iliev '01 (agent-tracking visualization; now a Ph.D student in CS at Dartmouth), Jason Kochel '99 (logistics), Michael Lewis '00 (now at America Online Inc.), Jieli Li '02, David Marmaros '01 (personal radio), Christopher Masone '02 (ad hoc routing and Solar; now a Ph.D student in CS at Dartmouth), Arun Mathias '01 (Serval, Solar), Neha Narula '02 (wireless networks, hand-held computers), Fred Reiss '00, Dan Scholnick '00, Matt Soldo '99 (communication), Jeffrey Steeves '99 (resource managers), Frederick Strathmeyer '02, Jennifer Turney (research intern 2002), Tarim Wasim '99 (sound support), Alik Widge '99 (Agent Scheme), Tiffany Wong '01 (wireless networks, hand-held computers), and Jeff Zimpleman '00 (performance measurements).

3 Research products

The primary product of our research was a series of technical papers published in the academic literature, and presented at academic conferences. We also released several pieces of software.

3.1 Papers published

All 86 of our publications are listed in the References section of this report, and on our web site.¹

We published 11 journal papers [AFALC02, BMC99, BMCI⁺03, BC00a, GCK⁺02, KGR02a, KCG⁺02, LR03, LPDR03, ALR03, MC01].

We presented 39 papers at refereed conferences and workshops [ACKR01, ARR00, ABC⁺03, BCC⁺03, BMCI⁺00, BKR01a, BKR98a, BKR99a, BKR00, BC00b, CK02a, CLK04, CK03, CK02d, CK01a, CPR03, CC03, CJ99, CJB99, CJ00, CM99, JC02, Gra00, GKP⁺01a, GGK⁺02, JCH03, JGWC99, JCC03, KSP⁺03, KG99b, KJG⁺00a, LAR03, LR02, LDR03, OC99, PR02, RSX01, XRS01].

We released 23 technical reports [BKR98c, BMCI⁺99, BKR99c, BKR97, Bre01b, CK02b, CK04b, CK04a, CK02c, CK01b, Che04b, CG00, Dub04, GCKR00, GKP⁺01b, GGK⁺01, KGR02b, KJG⁺00b, KCG⁺00, LRR02, MK02, Whi02a, Whi02b].

We contributed 4 book chapters [BKR⁺01b, BGM⁺99, GCKR02, GKCR98].

Finally, we wrote 8 other papers, theses, and posters [BKR98b, BKR99b, CK04c, Che04a, KG99a, LPRR02, Li04, Sun00].

Furthermore, 5 students completed a M.S. or Ph.D thesis in the project [Bre01a, Che04a, Li04, Moi98, Sun00], and 3 undergraduates completed a Senior Honors thesis [CG00, Dub04, Whi02a], and we applied for a patent on a technique for the encrypted execution of encrypted programs.

Since most of the technical reports were ultimately published as conference or journal papers, and some of the conference papers were expanded and published as journal papers, there are not 86 unique papers represented above; nonetheless, there are 50 peer-reviewed publications and overall it is a substantial record.

¹<http://agent.cs.dartmouth.edu/CoABS/papers/>.

These papers roughly fall into a few categories, which we summarize below.

3.1.1 Mobile agents.

The Dartmouth Agents, or D'Agents, system is a mobile-agent system that is distinguished by its support for strong mobility (so that an agent can migrate to a new machine at any point during its execution, implicitly carrying its state), its support for multiple agent languages (so that the agent programmer can select an appropriate language for the application), and its high performance (so that mobile-agent performance can be compared in a meaningful way with the performance of traditional client/server solutions). Although work on D'Agents began in 1994 as part of Gray's Ph.D thesis, we improved the performance, security and fault tolerance of D'Agents under the CoABS program [GCKR02, GCKR00, GCK⁺02, GKCR98], considered the future of mobile agents and mobile code [KGR02a, KGR02b, KG99b, KG99a], adapted a large existing application of D'Agents [Gra00] for use in the CoAX experiment; and used D'Agents as the starting point for all of our higher-level agent work.

This higher-level work can be divided into four categories. First, one weakness of existing mobile-agent systems was that they do not interoperate. An agent in one agent system can not communicate with the agents of or migrate to an agent platform of another agent system. Dartmouth, in cooperation with CoABS participants from Lockheed Martin and the University of Western Florida, developed the Grid Mobile-Agent System (GMAS), a set of standardized interfaces that allowed an agent conforming to that interface to function properly inside multiple agent systems [GGK⁺02, GGK⁺01]. GMAS was incorporated into the CoABS Grid software package.

Second, it is important to understand how the performance of mobile agents compares with that of traditional client/server solutions so that a developer can select the most appropriate implementation strategy for their application. Dartmouth, independently and in cooperation with the same set of partners, undertook a series of experiments that compared mobile-agent and client-server performance for information-processing applications (i.e., applications where information must be retrieved from multiple remote sites and then correlated for human use) [GCK⁺02, GKP⁺01a, GKP⁺01b, KCG⁺02, KJG⁺00a, KJG⁺00b, KCG⁺00]. Although many of the results conform to intuition (e.g., mobile agents are a better choice when available network bandwidth is low), the analysis of quantitative experimental results filled an important gap in available mobile-agent performance studies. As a companion effort, Dartmouth also examined the difficulties of simulating mobile-agent performance accurately enough to make performance predictions, and developed an accurate simulation package for information-processing applications [Dub04].

Third, many mobile agents have some choice as to which remote hosts to visit and in which order. Scheduling mobile-agent migration in a way that minimizes bandwidth and other resource use, therefore, is both feasible and attractive. Dartmouth considered several variations of the scheduling problem, and developed an algorithm for scheduling sequential agent-based access to remote resources [MC01, CM99, Moi98], several algorithms for scheduling parallel agent-based access to remote resources [RSX01, XRS01], and a graphical tool for allowing a human developer to quickly layout agent-based workflow applications [Sun00].

Finally, mobile agents can be much smaller if they dynamically find and use public "code libraries" rather than carrying all of their code with them. The dynamic use of third-party code, however, begs the question of whether the code actually performs the advertised computation. *Functional validation* is the task of verifying that a third-party package is actually computing what it says it is, and can be implemented as an extension to agent broker and matchmaker services (which, by themselves, cannot guarantee *functional* interoperability between agents written by different communities). We proposed a mathematical theory of such validation, and developed initial validation algorithms derived from this theory [CJ99, CJB99].

Overall, there were 26 papers in this category [CM99, CJ99, CJB99, Dub04, GCKR02, GCKR00, Gra00, GKP⁺01a, GKP⁺01b, GKCR98, GCK⁺02, GGK⁺02, GGK⁺01, KGR02a, KGR02b, KG99b, KG99a, KCG⁺02, KJG⁺00a, KJG⁺00b, KCG⁺00, MC01, Moi98, RSX01, Sun00, XRS01].

3.1.2 Market-based resource control.

We studied the use of markets to distributively allocate computational resources among software agents in computational systems. Markets proved useful in providing incentives to participate in distributed applications, prioritizing tasks, and adding additional fault tolerance. Additionally, we derived structures to facilitate planning and control the degree of private information disclosure, and structures that allow agents to exchange volatility with performance.

Incentives. As distributed systems get larger, the number of principals that contribute to an application increases and a particular resource owner may not immediately realize benefit from system participation. By charging resource usage to a software agent's account, a resource owner can extract value from the use of its wares; value that could be exchanged for services at other sites or for another, possibly legal tender, resource [BKR98a, BKR99a].

Since each agent has a limited endowment with which to acquire resources, an agent's potential to wreak havoc on the network is limited. A long-lived agent must either choose to act at times with low resource-usage demand, or receive a large endowment from its owner. The budget constraint provides a degree of fault tolerance against malicious and buggy code [BKR97, BKR98a, BKR99a, Bre01a].

Finally, price systems allow flexible discrimination among agents. More well-endowed agents compute more quickly than less-endowed agents, especially during times of resource contention. Our experiments show that systems using our resource allocation can effectively serve wealthy agents, even when total load far exceeds capacity [BKR99c, BMcI⁺99, BMcI⁺00, BKR⁺01b, Bre01a, BMcI⁺03].

Structures. Providing market structure only partially solves resource-allocation problems; an agent must be able to interact with the market and plan to complete its goals. Towards this end, we studied many different auction schemes in which agents bid for computational resources [BKR98b, BKR99c, BMcI⁺99, BMcI⁺00, BKR⁺01b, BKR01a, Bre01a]. We found a tradeoff in the information regarding preferences agents disclosed and efficiency of the market and in the complexity required to calculate prices [BKR01a, Bre01a].

Most of the market mechanisms we studied were demand driven. An agent purchased resources immediately prior to use. To alleviate risk, however, we instrumented call options in our market where agents could purchase a contract to guarantee computation at a fixed price in the future. We extended the Cox, Ross, Rubinstein option-pricing model to allow agents to integrate demand and reservation-based computing according to their preference towards risk [BKR00, Bre01a].

Each of the planning algorithms we implemented relied on estimates of an agent's planned consumption. In real-life applications, resource use is difficult to forecast, so we tested our algorithms using flawed estimates to find that our algorithms adapted an agent's plans as the estimation errors became more apparent [Bre01a, BMcI⁺03]. Finally, we implemented our resource-allocation and planning algorithms as part of a Linux real-time kernel scheduling process to regulate mobile processes [CG00].

There are 17 papers in this category [BKR98b, BKR98c, BMcI⁺00, BKR⁺01b, BMcI⁺99, BKR01a, BMcI⁺03, BKR99b, BKR99c, BKR98a, BKR97, BKR99a, BKR00, Bre01a, Bre01b, CG00].

3.1.3 Mobile computing.

Our research in the CoABS program led us to think about new ways to use mobile agents, and mobile code in general, to support information gathering, processing, and dissemination. In our efforts to model and simulate the performance benefits of mobile code in applications where a large number of users (or applications) need to obtain filtered information from an ongoing stream of data [KCG⁺02, KJG⁺00a, KJG⁺00b, KCG⁺00], we began to think about ways to generalize this methodology. Indeed, the foundational ideas

were developed during a stroll with colleagues outside one of the DARPA PI meetings. The result was the “Solar” project, with 16 papers resulting [CK02a, CK02b, CK04c, CK04b, CLK04, CK03, CK04a, CK02d, CK02c, CK01a, CK01b, Che04a, Che04b, MK02, Whi02a, Whi02b]. One Ph.D and several undergraduate theses resulted; another two Ph.D theses and one M.S. thesis are expected. Solar is middleware designed to support context-aware applications, that is, applications that want to dynamically adjust their behavior to suit their changing environment. In many pervasive-computing applications, the environment includes the physical world around the user and the application; the application must locate and use information coming from physical sensors. In command-and-control applications, the application needs to gather information from a wide variety of information sources, both physical and virtual, both near and far, and to filter and correlate the information to provide situational awareness. Solar is designed to support both applications well, with many of the same goals as the Joint Battlespace Initiative’s own middleware. Solar allows the application to deploy mobile code in the form of *operators* that use context-sensitive attribute-based names to request the desired sensor information sources, subscribe to those sources, and then pre-process the data to produce the desired information stream to the application(s) that need it. The use of mobile code allows us to shift the computation away from the user device and into the infrastructure (which has abundant power, bandwidth, and better security). The mobile code can also migrate from host to host within the infrastructure, to better optimize the computation and communication loads [CLK04]. The Solar project has investigated the deep systems issues in developing such middleware, including naming (for resource discovery) [CK03], load balancing (migrating operators), load reduction (when information flow exceeds the capacity of the system) [CK04a], reliability (monitoring and recovering from failure of operators and hosts), security (controlling access to sensitive information in the flow) [MK02], and scalability. We built the entire software system, measured its performance, and demonstrated its value with several applications [Che04b].

3.1.4 Wireless networks: Using mobile agents for routing.

We developed an application of mobile agents that supports communication in the presence of disconnections in mobile wireless computer networks. The main advantage of using mobile agents for communication in ad-hoc networks is that they can function as “wrappers” on messages. The mobile agent wrapper (called an active message) provides a certain level of autonomy for messages and allows them to reside at intermediate points in the network. This enables a message to propagate itself to the destination incrementally, which is an advantage over traditional message transmission approaches in which the entire path from the starting location to the destination must be available. Thus, the communication protocol we propose is an application-layer protocol (rather than a network-layer protocol) [OC99].

When the network cannot route a message to its destination due to a network partition, it can use our algorithm to determine new positions (and trajectories to those positions) for the hosts relaying the message. For example, in a tactical robotic network where a team of robots is deployed to perform sensing tasks in a remote or hazardous environment, the message routing program could suggest trajectory modifications for the team, while the individual robots can decide the ultimate host trajectories. Relevant papers include [CPR03, LR03, LR02, Li04].

Our approach guarantees message transmission in minimal time. Since this approach asks mobile hosts to actively modify their trajectories to transmit messages, we develop algorithms that minimize the trajectory modifications under two different assumptions: (a) the movements of all the nodes in the system are known and (b) the movements of the hosts in the system are not known. We showed how the information about the motion of the destination host can be used to determine how the message can be sent by the cooperation of the intermediate hosts. Given an ad hoc network of mobile computers where the trajectory of each node is approximately known, our algorithm computes a trajectory for sending a message from host A to host B by recruiting intermediate hosts to help. In this context, recruiting means asking intermediate hosts to change their trajectory to complete a routing path between hosts A and B. We would like to minimize the trajectory

modifications while getting the message across as fast as possible.

The 17 papers in this category also include several other papers related to routing, tracking, and interactive protocols for ad hoc networks of sensor devices [ABC⁺03, CC03, JC02, KSP⁺03, LAR03, LPDR03, LPRR02, LDR03, LRR02, ALR03, PR02].

3.1.5 Mobile agents for information retrieval.

We used transportable agents primarily for distributed information access, in which a distributed collection of corpora is searched based on a query and the results extracted from each site are fused in a coherent picture. The main advantages of using agents in distributed information access are flexibility and performance. With agents, distributed collections can provide primitive operations rather than all possible search operations. An agent can combine these primitives into efficient, multi-step searches. By moving a small computation to the location of the data (with transportable agents), the network traffic and overall computation time is reduced.

We built information-access agents that interface with the well-known “Smart” information retrieval system. The Smart system is a successful statistical information-retrieval system that uses the vector-space model to measure the textual similarity between documents. The idea of the vector-space model is that each word that occurs in a collection defines an axis in the space of all words in the collection. A document is represented as a weighted vector in this space. The premise of this system is that documents that use the same words map to neighboring points and that statistics capture content similarity. Our “star” algorithm organizes a document collection into clusters that are naturally induced by the topic structure of collection, via a computationally efficient cover by dense subgraphs [ARR00].

Our data is a distributed collection of document repositories, each running an information-retrieval system. In our prototype, each collection consists of computer science technical reports. For a given query, an information agent visits a sequence of sites; at each site, it interacts with the local Smart agent to search the local collection. The results retrieved are brought home, or used as relevance feedback to refine the query. We consider the advantages and disadvantages of mobile agents for this sort of task, and develop planning algorithms suitable to minimize overhead [BGM⁺99].

We also conducted a series of related, but unpublished, experiments to measure the scalability and performance of persistent queries in a large document collection. Our Standing Query Server (SQS) received keyword-based queries from clients, performed an initial database search for the first 50 matching documents, clustered the resulting documents using an implementation of the “star clustering” algorithm mentioned above, and then stored the query. When new documents arrived in the database, the system ran each stored query over the new documents, and recalculated the document clusters. If a new document joined one of the clusters selected as most relevant by the client, and was above a certain relevance threshold, the client was notified and could retrieve the document. We ran experiments to discover the effect of the relevance threshold, database size, and number of standing queries on the performance of the system, and determined that the algorithm was consuming significant memory resources to cache associations. In fact, the memory used was more than the combined size of the documents themselves, leading to performance tradeoffs between the number of new documents added and the number of standing queries that could be supported. Overall, the performance did not appear to be practical for the quickly-growing document collections that we targeted.

3.1.6 Other papers.

There are nine other papers that do not readily fit into any of the categories above.

Our work with mobile agents, sensor networks, and information retrieval led us to study the challenge of hypothesis tracking; one paper investigates a fuzzy version of the multiple hypothesis tracking (MHT)

problem [AFALC02]. Another paper begins a broad new approach to this and many related problems, using a new approach we call process query systems [BCC⁺03].

Another paper sketches out the important directions in scientific and distributed computing, based in part on our experience with the other research represented here; this paper was published in the premier issue of a new IEEE magazine [BMC99].

Two papers attempt to measure how quickly the content on the WWW changes [BC00b, BC00a], with applications to other information-monitoring applications. For example, the Infrastructure Web explores methods for monitoring critical infrastructures [CJ00].

To be able to monitor a diverse range of sensor inputs, it is important to be able to understand their semantics, or at least to allow different components to interoperate through semantic markup [JCH03, JCC03].

One paper on Q-learning represents our effort to address the challenges of mobile-agent planning [JGWC99].

3.2 Software

We updated and released the source code for our D'Agents mobile-agent system. We have reports of this software being used for research in other universities and corporate labs.

We improved the D'Agents performance by multi-threading the server code and by pre-launching interpreters. We also extended our Java support to run multiple agents within the same Java virtual machine, reducing overhead. We extended our communication mechanisms to allow the transmission of arbitrary binary data, and added compression capabilities to minimize the size of transmitted messages.

We wrote a "yellow-pages" service to allow agents to locate desired services by their attributes. This feature was later subsumed by the CoABS Grid's discovery service.

We developed an infrastructure for resource management, in which mobile agents needing access to local machine resources are limited by policies implemented in separate resource manager agents. Our first resource manager limited access to the screen, so that mobile agents could only open windows on the screen when allowed by the manager. This work later evolved into the market-based control approach, described below, when we started to work with more consumable resources like CPU time.

We developed a primitive mobile-agent system for the Palm III hand-held computer. Agents comprised of pre-compiled Palm code and interpreted Tcl code could jump on and off Palms. The Palm code ran when the agent was on the Palm platform, the Tcl code ran when the agent was on a Unix platform. The two parts could communicate through state variables brought along by the jump operation. They could locate Palms by names chosen by the Palm user, through our Yellow Page server. Later, we developed an independent mobile-agent platform for Windows CE, and several simple applications.

We developed a scalable document server, called Serval, and prototyped it on our 24-node Linux cluster, called Serengeti. This document server allowed access by either RPC or mobile agents, and was a platform for testing the scalability of our agent system compared to that of traditional (RPC) mechanisms. It also allowed us to better integrate C++ and Java agents, and drove many new performance enhancements in D'Agents.

As part of the CoABS Grid effort, we co-developed the Grid Mobile Agent System (GMAS) with Lockheed Martin's ATL and the University of Western Florida's IHMC. GMAS was a major extension to the Grid that defined a Java API for mobile Grid agents. Through proxy-based launchers and bridges, a Grid agent could launch a mobile agent into any Java-based mobile-agent system that supported this API, and could communicate with the mobile agent or any other agent on that system. The mobile agent also could migrate *between* different agent systems, as long as all of the systems supported the API. As a test of GMAS functionality, we added support for the GMAS API to D'Agents, EMAA, and NOMADS, the mobile-agent systems of Dartmouth, Lockheed Martin, and UWF respectively, and demonstrated an application agent migrating sequentially through the three different agent platforms. This same agent was used to provide

medical-monitoring support in the CoAX effort, a joint experiment and technology demonstration designed to explore the efficacy of the CoABS software in coalition applications (see Section 5 for more information). With GMAS integrated into the main CoABS Grid distribution, Grid applications easily can take advantage of the bandwidth and latency advantages of mobile agents.

4 Demonstrations and Presentations

4.1 Demonstrations

We demonstrated prototypes of our technology in many venues.

Our most significant demonstrations were part of the “Mobility TIE,” a group of CoABS-funded researchers primarily at Dartmouth College, Lockheed-Martin Advanced Technology Lab, and the University of Western Florida’s Institute for Human and Machine Cognition. That group had three major efforts:

1. to add mobility to the CoABS Grid framework. The result was the Grid Mobile-Agent System (GMAS), which was demonstrated (at least) at the CoABS Transition Exhibition in Miami and the CoABS PI meeting in Virginia.
2. to evaluate the scalability of mobile-agent systems, by comparing D’Agents with those from other CoABS-funded efforts. This head-to-head comparison, although not a live demo, led to extensive experimental results that were presented at CoABS meetings and elsewhere. (See the Papers and Presentations section for more information.)
3. to consider resource-control frameworks for mobile-agent systems. We developed a common API to specify resource needs and limits.

In the context of the Mobility TIE we participated in two Fleet Battle Experiments, gaining data that we integrated into our research [KCG⁺02].

We were also a major player in the CoAX demonstrations, in which the CoABS Grid, our GMAS, and other CoABS technologies were part of a series of Coalition Forces demonstrations. We demonstrated our GMAS technology, and our mobile-agent-based medical monitoring application, at the CoAX meeting in October 2002.

We also integrated CoABS technology into our work in another DoD-funded effort, a MURI we call ActComm.² In August 2002 we held a major demonstration and presentation of work in that project. These demonstrations included our resource-control algorithms running inside D’Agents and an enhanced version of D’Agents, made more scalable through CoABS scalability research. We also demonstrated this technology at the CoABS Science Fair.

In more recent work, Paul Thompson (a Dartmouth collaborator of George Cybenko) demonstrated “Dynamic Integration of Distributed Semantic Services: Infrastructure for Process Queries and Question Answering” at Human Language Technologies 2003.³

4.2 Presentations

In addition to presenting 39 conference and workshop papers, above, our team gave several invited lectures at conferences, universities, industry labs, and government groups.

²See <http://actcomm.thayer.dartmouth.edu/>.

³<http://www.sims.berkeley.edu/research/conferences/hlt-naacl03/demos.html>

Conferences. In no particular order.

- Cybenko was a keynote speaker at the May 2003 SIAM Conference on Data Mining, in San Francisco.
- Cybenko was a keynote speaker at the July 2003 SIAM Conference on Applied Linear Algebra at College of William and Mary.
- Cybenko and Gray gave a tutorial on Mobile Agents at the IPPS/SPDP conference in Puerto Rico.
- Cybenko presented *Technology and Globalization* at the Salzburg Seminar.
- Kotz gave a talk at the Dartmouth Workshop on Transportable Agents.
- Cybenko gave the wrapup talk at the Dartmouth Workshop on Transportable Agents.
- Bredin (student) gave a poster at ASA/MA '99.
- Gray gave the Keynote presentation at PAAM 2000.
- Gray presented a tutorial at PAAM 2000, in England. The tutorial covered a broad range of mobile agents research.
- We had a poster presented at AA2001.
- Rus gave a talk at the DIMACS Workshop on Pervasive Networking, 2001.
- Cybenko gave an invited talk at the 2002 SIAM Southeast Regional Meeting, on Security of Mobile Code and Quantum Computing Models.
- Kotz presented an invited keynote lecture at "Mobile Agents 2002" called "Beyond Mobile Agents: Applications of Mobile Code to Pervasive Computing."
- Chen (student) presented a work-in-progress talk at OSDI (Operating Systems Design and Implementation), 2002.
- Minami (student) presented a poster at Mobisys 2003.
- Chen (student) presented a poster at Mobisys 2003.

Universities. In no particular order.

- Cybenko gave talk at U Tennessee.
- Kotz gave a talk at University of Vienna.
- Rus gave a talk at UC San Diego.
- Cybenko gave a lecture in the "Computational Science and Engineering" program at Florida State University.
- Cybenko gave a talk at Memorial University, St. Johns, Newfoundland, Canada.
- student Jon Bredin, on graduating, gave numerous presentations of his dissertation work while interviewing at universities around the country. His work is entirely about resource control in mobile-agent systems, funded by DARPA CoABS.

- Cybenko gave a talk at Purdue University, Samuel Conte Lecture.
- Cybenko gave a talk at the University of Kentucky, Center for Computational Science.
- Kotz gave a talk at Georgia Tech.
- Cybenko gave a talk a Harvard about information flow in sensor networks.
- Staff programmer Arne Grimstrup gave a talk at the University of Manitoba CS department.

Industry labs. In no particular order.

- Rus gave a talk at Caterpillar.
- Kotz presented the concept of the CoABS grid, and several of our research projects, in a colloquium at IBM T.J. Watson Research Center.
- Kotz gave a talk about D’Agents performance at Mitsubishi Electric Research Lab.
- Cybenko gave a talk at ALPHATECH Burlington, MA.

Government. In no particular order.

- Rus gave talk at Rome Labs (Joe Cavano et al.).
- Cybenko gave an overview of agent research at AFOSR.
- Rus spoke at the AFoSR MURI review, Ithaca, NY.
- Kotz gave a talk in the AFRL seminar series.
- Bredin (student) gave a talk in the AFRL seminar series.
- (in absentia) preparation of an SAB poster at the request of AFRL.
- Gray gave an Overview of Mobile Agent Scalability at the Armed Forces Communication and Electronics Association (AFCEA) Erie Canal Chapter Sponsor for Information Superiority Conference, 2001.
- Cybenko spoke at an IEEE Information Theory Seminar at Lincoln Labs.
- Cybenko presented a seminar on the security of mobile code at the IDA Center for Computing Systems.
- Cybenko presented a status report on the ISAT Mobility and Security Study at IDA, Alexandria, VA.
- Rus attended an NSF workshop on mathematical questions in robotics and led the discussion on distributed robotics issues.
- Rus gave a talk at NRL.
- Cybenko gave a talk about “Monitoring distributed information systems” at CNR (the Italian National Research Council), Pisa Italy in March 2001.⁴

⁴<http://www.cnr.it/>

- Cybenko spoke about Semantic Services for C4ISR Systems at Agent Technology for the Objective Force, US Army Workshop, Potomac, MD, 2002.
- Kotz, Gray, Cybenko, Rus each presented progress reports on CoABS-related research at the semianual meeting of the “Actcomm” project, a DoD MURI funded effort. Representatives of ARL, ONR, and AFoSR were in attendance, as well as academic and industrial participants.
- Cybenko spoke about the Detection and Control of Malicious Code Attacks, at the DARPA Industry Day Workshop on Dynamic Quarantine, March 2003, Arlington VA.

CoABS meetings. In no particular order.

- Cybenko presented the Scalability TIE Overview at DARPA CoABS workshop in Las Vegas.
- Kotz spoke at the CoABS workshop in Northampton.
- Cybenko spoke at the CoABS workshop in Northampton.
- Gray spoke at the CoABS workshop in Northampton.
- Cybenko gave a talk at DARPA CoABS Science Fair.
- Cybenko presented our modeling work at the March 2000 CoABS workshop.
- Kotz presented the Grid/mobility design at the March 2000 CoABS workshop.
- Kotz presented the Mobility TIE update at the DARPA CoABS PI meeting, Miami.
- We presented eight posters at the DARPA CoABS Transition Exhibition, Miami.
 - Dartmouth CoABS overview.
 - D’Agents overview.
 - Dartmouth ACTCOMM overview.
 - Dartmouth persistent-query experimental results.
 - Dartmouth modeling/simulation results.
 - Mobility TIE scalability experiments.
 - Mobility TIE Grid-mobile-agent system.
 - Fleet Battle Experiment, plan for 2001.
- Kotz gave an update on our work at the CoABS PI workshop, Nashua NH.
- Gray and Cybenko participated in the CoABS review, in Virginia, 2001.
- Cybenko attended the NASA Workshop on long duration space flight, Atlanta GA (this work uses mobile agents to manage spacecraft systems and crew functions).
- Gray presented an update on our progress and our FY02 plans at the CoABS workshop in Virginia.

Courses. We incorporated our work into a new course, “Wireless networks and hand-held computers,” to train undergraduate and graduate students in many of the key background fields. We later enticed the best students into our research team, either as hired programmers or honors research students.

5 Technology Transfer

Our CoABS efforts had a significant impact on our other DoD-funded research efforts, and we were able to use our results and prototypes to support that research.

As one example, our mobile-agent technology was used in an application for medical monitoring of victims on a battlefield, funded by the U.S. Army CECOM, and a related application for medical monitoring of astronauts in long-duration space flight, funded by the NASA Institute for Advanced Concepts. The Grid was used to provide interoperability between D'Agents and other agent systems, and was used for the Guardian Angel system developed by Lockheed Martin. This project was led by Dr. Sue McGrath, a collaborator who learned of our technology while at Lockheed Martin ATL and then joined Dartmouth College to pursue the research here.

We were deeply involved with the CoAX TIE. Gray and S. McGrath provided a version of the CECOM medical-monitoring system for use in this TIE. This system uses the Grid Mobile Agent System (GMAS) to send monitoring agents to the location(s) of injured soldiers and sailors, allowing medics from one country to efficiently monitor wounded sailors from another country. We demonstrated the final version, with a (simple) graphical user interface for the medics, as part of the final CoAX presentations in October 2002. In the CoAX presentation, the system was used to monitor the health of casualties on board an Australian ship, *HMAS Coonawarra*, which was damaged during a submarine attack. Representatives from the Marine Corps expressed significant interest in the system during the Tech Fair following the presentations; McGrath and Gray are following up with these representatives.

We also made several efforts to raise awareness of our results and technologies within the DoD world and the corporate world, above and beyond the impact of our publications, presentations at conferences, and presentations at university, corporate, and government labs. In particular, we sent a high-level summary of our research, with pointers to detailed results, to a hundred or so key players in the DoD world.

On many occasions we discussed transferring our technology...

- with BAE Systems (British AeroSpace) about using Dartmouth's agent and security work in their advanced communications systems.
- with the Director of Research at the FAA about transitioning some of our work to that setting.
- with Jeff Hughes of Wright Patterson AFB on a potential OSD Secure Applications Initiative.
- with the Grid Forum.⁵ Kotz participated in the first meeting of the Grid Forum, a large nationwide group that is trying to coordinate the interconnection of several metacomputing efforts, from NASA, NSF, DOE, DOD, and numerous others. Kotz was there to investigate potential technology transfer from the CoABS "Grid" to and from the other grid-like projects. Cybenko had earlier met with NASA Ames to discuss coordination of the CoABS Grid with NASA's Information Power Grid. Visited with Dennis Gannon and Bill Johnston about possible relationships between the two efforts.

6 Community service and advisory functions

We believe that it is important for us to contribute to the broader research community, by helping to organize conferences, sitting on review boards, and the like. We also made an effort to support the needs of federal programs. Indeed, we also believe that this service helps to publicize the efforts of the CoABS program and to extend the value of its research.

Here is a partial listing of our community service, in no particular order.

⁵<http://www.gridforum.org>

Conference committee service

- Kotz was Program Chair for the conference “Agent Systems and Applications / Mobile Agents” (ASA/MA 2000), held in Zurich in September 2000.
- Kotz, Gray, and Rus were co-chairs of the Dartmouth Workshop on Transportable Agents (DWTa 2000), also held in Zurich. We invited 12 top researchers to consider the question “What are the most important research directions in the mobile-agent community, in order to have an impact outside our community?” We recorded the discussion and published a summary in a well-known IEEE publication.
- We helped to organize Autonomous Agents 99 (AA99).
- We helped to organize a workshop on Scalable Agent Systems at AA99.
- We helped to organize a workshop on Mobile-Agent Systems at AA99.
- We played a major role in the organization of “Agent Systems and Applications / Mobile Agents” (ASA/MA99), and we organized the concurrent Dartmouth Workshop on Transportable Agents (DWTa99).
- Rus was on the Program Committee for Autonomous Agents 2000.
- Rus was on the senior program committee for AA2001.
- Cybenko was a ICDCS “Distributed Agent Systems” PC Member
- Rus was general chair of ISER 2000.
- Kotz was General Chair of MA2001.
- Gray was Treasurer of MA2001.
- Gray was on the Program Committee of MA2001.
- Gray was on the program committee for of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-2003).

Journal service

- Kotz was co-Editor of a special issue on High-Performance Agent Systems in *Concurrency Practice and Experience*.
- Cybenko was on the the IEEE Computer editorial board.
- Cybenko was elected to the IEEE Computer Society Board of Governors, 2002-2004 term.
- Cybenko was named the founding Editor-in-Chief of *IEEE Security and Privacy*, which published its first issue in February 2003.
- Kotz was an associate editor for ACM SIGMOBILE’s *Mobile Computing and Communications Review*.

Government service

- Kotz was named Director of Research and Development for the Institute for Security Technology Studies, a research institute funded by DHS/ODP, at Dartmouth College.
- Cybenko was on the organizing committee for the DARPA Workshop on Trustworthy Computing in Dynamic Environments, Sept 17-18, 2002, with Anup Ghosh DARPA PM.
- Cybenko co-chaired a DARPA ISAT Study on mobility and security, which began in 2000.
- Cybenko was invited to be on the DoD Technology area review and assessment committee, held at Rome Labs.
- George Cybenko participated in the Air Force Information Directory Strategic Planning meeting.
- Cybenko worked with Nort Fowler to define the concept of “Infospheric Science” or “Infospherics,” which is the scientific basis for designing large, heterogeneous, extensible systems. The report, with Bob Herklotz’s and Nort Fowler’s input and feedback, was based on a workshop held in July 2000.
- Cybenko was invited to act as a DOD Technology Area Assessment and Review Panelist in 2002 for the DoD IT research area (Ft. Monmouth, 2002).
- Cybenko was on the Executive Committee of the Northeast Regional Research Center of the intelligence community’s Advanced Research and Development Activity (ARDA).
- Cybenko was invited to serve on 2002 TARA IT Panel, April 2002, Ft. Monmouth NJ.
- Cybenko was nominated to the Air Force Science Advisory Board by Dr. Alex Levis, AF Chief Scientist.
- Cybenko participated in a DARPA IXO Study Panel on Information Integration.
- Cybenko and Rus went to AFRL/Rome Labs for the Air Force Science Advisory Board Review.
- Cybenko was the Editor of IEEE Security and Privacy Supplement to IEEE Computer Magazine, which appeared approx. April 2002.
- Cybenko chaired the IEEE Security and Privacy Task Force.
- Dartmouth hosted the AFOSR/ARL-IF Strategic Directions Workshop in August 2002.
- Cybenko served on the IEEE-USA R&D Policy Committee, which provides expert advice and congressional testimony about computing and security legislation.
- Dartmouth hosted the AFOSR/ARL-IF Strategic Directions Workshop in August 2002.
- Cybenko was the Chief Scientist-at-Large for the Office of Secretary of Defense’s Software Protection Initiative managed by AFRL WP SD. Jeff Hughes was the PM.
- Cybenko visited Rome Labs for a planning session about the Joint Battlespace Infosphere, and later was a member of the AFRL JBI Steering Group.
- Cybenko presented the ISAT report to the DARPA director in a meeting at DARPA.
- Cybenko participated in the Air Force Information Directory Strategic Planning meeting.

7 Conclusions and lessons learned

We came to many conclusions; the details are in our papers.

- Mobile code, and often mobile agents, are an efficient mechanism for information filtering and retrieval applications. We learned that a carefully constructed implementation can be scalable and can outperform a traditional remote procedure call approach, in situations where the communication bandwidth is limited and the remote execution of application logic can reduce the communication needed. We conducted the largest scalability experiments of any mobile-agent system, and indeed directly compared the scalability three mobile-agent systems.
- Market-based control is an effective method for balancing load in a distributed system, with self-interested mobile agents seeking out the lowest price to accomplish their tasks. We devised auction-based schemes to support this idea.
- Mobile agents can be used to carry messages through an ad hoc wireless network that is prone to disconnection. Indeed, the agent can ask the network nodes to move (physically) so that the message can reach a currently unreachable portion of the network. We demonstrated algorithms that make this possible.
- The ideas behind mobile agents can be generalized to our Solar framework that can support distributed information filtering, processing, and dissemination. Here, mobile code allows the computation to move closer to the data, can be shared to allow for scalability in the face of large numbers of applications desiring the same information stream, and can be rearranged to balance load.
- And many others.

Overall, we contributed substantially to a better understanding of mobile-agent systems and related areas of ad hoc wireless networks, sensor networks, information retrieval, and middleware for context-aware computing and sensor-information processing. Building on our experience in the field we also wrote and presented forward-looking articles on mobile-agent research.

Our research resulted in 86 publications, numerous demonstrations of research prototypes at government and industry labs, transfer of our technology to other research efforts (including contributions to the CoABS Grid software base), dozens of research students trained, and extensive contributions to both the research and government communities.

References

- [ABC⁺03] Javed Aslam, Zack Butler, Florin Constantin, Valentino Crespi, George Cybenko, and Daniela Rus. [Tracking a moving object with a binary sensor network](#). In *Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 150–161, Los Angeles, CA, November 2003. ACM Press.
Abstract: In this paper we examine the role of very simple and noisy sensors for the tracking problem. We propose a binary sensor model, where each sensor’s value is converted reliably to one bit of information only: whether the object is moving toward the sensor or away from the sensor. We show that a network of binary sensors has geometric properties that can be used to develop a solution for tracking with binary sensors and present resulting algorithms and simulation experiments. We develop a particle filtering style algorithm for target tracking using such minimalist sensors. We present an analysis of fundamental tracking limitation under this sensor model, and show how this limitation can be overcome through the use of a single bit of proximity information at each sensor node. Our extensive simulations show low error that decreases with sensor density.
- [ACKR01] Jay Aslam, Marco Cremonini, David Kotz, and Daniela Rus. [Using mobile agents for analyzing intrusion in computer networks](#). In *Proceedings of the Workshop on Mobile Object Systems at ECOOP 2001*, July 2001.
- [AFALC02] Santiago Aja-Fernández, Carlos Alberola-López, and George V. Cybenko. [A fuzzy MHT algorithm applied to text-based information tracking](#). *IEEE Transactions on Fuzzy Systems*, 10:360–374, June 2002.
Abstract: We carry out a detailed analysis of a fuzzy version of Reid’s classical multiple hypothesis tracking (MHT) algorithm. Our fuzzy version is based on well-known fuzzy feedback systems, but the fact that the system we describe is specialized for likelihood discrimination makes this study particularly novel. We discuss several techniques for rule activation. One of them, namely the sum-product, seems particularly useful for likelihood management and its linearity makes it tractable for further analysis. Our analysis is performed in two stages: 1) we demonstrate that, with appropriately chosen rules, our system can discriminate the correct hypothesis; and 2) the steady-state behavior with a constant input is characterized analytically. This enables us to establish the optimality of the sum-product method and it also gives a simple procedure to predict the system’s behavior as a function of the rule base. We believe this fact can be used to devise a simple procedure for fine-tuning the rule base according to the system designer’s needs. The application driving our fuzzy MHT implementation and analysis is the tracking of natural language text-based messages. This application is used as an example throughout the paper.
- [ALR03] Javed Aslam, Qun Li, and Daniela Rus. [Three power-aware routing algorithms for sensor networks](#). *Wireless Communications and Mobile Computing*, 3(2):187–208, March 2003.
Abstract: This paper discusses online power-aware routing in large wireless ad-hoc networks (especially sensor networks) for applications where the message sequence is not known. We seek to optimize the lifetime of the network. We show that online power-aware routing does not have a constant competitive ratio to the off-line optimal algorithm. We develop an approximation algorithm called $\max\text{-min}$ zP_{\min} that has a good empirical competitive ratio. To ensure scalability, we introduce a second online algorithm for power-aware routing. This hierarchical algorithm is called zone-based routing. Our experiments show that its performance is quite good. Finally, we describe a distributed version of this algorithm that does not depend on any centralization.
- [ARR00] Jay Aslam, Fred Reiss, and Daniela Rus. [Scalable information organization](#). In *Proceedings of RIAO 2000 (Content-based information access)*, pages 1128–1138, Paris, France, April 2000. CID-CASIS.
Abstract: We present three scalable extensions of the star algorithm for information organization that use sampling. The star algorithm organizes a document collection into clusters that are naturally induced by the topic structure of collection, via a computationally efficient cover by dense subgraphs. We also provide supporting data from extensive experiments.

- [BC00a] Brian Brewington and George Cybenko. [Keeping up with the changing web.](#) *IEEE Computer*, 33(5):52–58, May 2000.
Abstract: Because information depreciates over time, keeping Web pages current presents new design challenges. This article quantifies what "current" means for Web search engines and estimates how often they must reindex the Web to keep current with its changing pages and structure.
 Most information—from a newspaper story to a temperature sensor measurement to a Web page—is dynamic. When monitoring an information source, when do our previous observations become stale and need refreshing? How can we schedule these refresh operations to satisfy a required level of currency without violating resource constraints—such as band-width or computing limitations on how much data can be observed in a given time?
 The authors investigate the trade-offs involved in monitoring dynamic information sources and discuss the Web in detail, estimating how fast documents change and exploring what constitutes a "current" Web index. For a simple class of Web-monitoring systems—search engines—they combine their idea of currency with actual measured data to estimate revisit rates.
- [BC00b] Brian E. Brewington and George Cybenko. [How fast is the web changing?](#) In *Proceedings of the WWW9 Conference*, Amsterdam, May 2000. W3C.
Abstract: Recent experiments and analysis suggest that there are about 800 million publicly-indexable web pages. However, unlike books in a traditional library, web pages continue to change even after they are initially published by their authors and indexed by search engines. This paper describes preliminary data on and statistical analysis of the frequency and nature of web page modifications. Using empirical models and a novel analytic metric of "up-to-dateness", we estimate the rate at which web search engines must re-index the web to remain current.
- [BCC⁺03] Vincent Berk, Wayne Chung, Valentino Crespi, George Cybenko, Robert Gray, Diego Hernando, Guofei Jiang, Han Li, and Yong Sheng. [Process query systems for surveillance and awareness.](#) In *Proceedings of the Seventh World Multiconference on Systemics, Cybernetics and Informatics (SCI 2003)*, Orlando, FL, July 2003. International Institute of Informatics and Systemics.
Abstract: Many surveillance and sensing applications involve the detection of dynamic processes. Examples include battlefield situation awareness (where the processes are vehicles and troop movements), computer and network security (where the processes are worms and other types of attacks), and homeland security (where the processes are terrorist financing, planning, recruiting, and attack-execution activities). A Process Query System (PQS) is a novel and powerful software front-end to a database or real-time sensing infrastructure that allows users to define processes at a high level of abstraction and submit process definitions as queries. We describe a current working implementation that has been used for vehicle tracking using an acoustic sensor network and for computer worm detection.
- [BGM⁺99] Brian Brewington, Robert Gray, Katsuhiko Moizumi, David Kotz, George Cybenko, and Daniela Rus. [Mobile agents for distributed information retrieval.](#) In Matthias Klusch, editor, *Intelligent Information Agents*, chapter 15, pages 355–395. Springer-Verlag, 1999.
Abstract: A mobile agent is an executing program that can migrate during execution from machine to machine in a heterogeneous network. On each machine, the agent interacts with stationary service agents and other resources to accomplish its task. Mobile agents are particularly attractive in distributed information-retrieval applications. By moving to the location of an information resource, the agent can search the resource locally, eliminating the transfer of intermediate results across the network and reducing end-to-end latency. In this chapter, we first discuss the strengths of mobile agents, and argue that although none of these strengths are unique to mobile agents, no competing technique shares all of them. Next, after surveying several representative mobile-agent systems, we examine one specific information-retrieval application, searching distributed collections of technical reports, and consider how mobile agents can be used to implement this application efficiently and easily. Then we spend the bulk of the chapter describing two planning services that allow mobile agents to deal with dynamic network environments and information resources: (1) planning algorithms that let an agent choose the best

migration path through the network, given its current task and the current network conditions, and (2) planning algorithms that tell an agent how to observe a changing set of documents in a way that detects changes as soon as possible while minimizing overhead. Finally, we consider the types of errors that can occur when information from multiple sources is merged and filtered, and argue that the structure of a mobile-agent application determines the extent to which these errors affect the final result.

- [BKR97] Jonathan Bredin, David Kotz, and Daniela Rus. [Market-based resource control for mobile agents](#). Technical Report PCS-TR97-326, Dept. of Computer Science, Dartmouth College, December 1997.
Abstract: Mobile agents are programs that can migrate from machine to machine in a heterogeneous, partially disconnected network. As mobile agents move across a network, they consume resources. We discuss a system for controlling the activities of mobile agents that uses electronic cash, a banking system, and a set of resource managers. We describe protocols for transactions between agents. We present fixed-pricing and dynamic-pricing policies for resources. We focus on and analyze the sealed-bid second-price auction as a mechanism for dynamic pricing.
- [BKR98a] Jonathan Bredin, David Kotz, and Daniela Rus. [Market-based resource control for mobile agents](#). In *Proceedings of the Second International Conference on Autonomous Agents*, pages 197–204. ACM Press, May 1998.
Abstract: Mobile agents are programs that can migrate from machine to machine in a heterogeneous, partially disconnected network. As mobile agents move across a network, they consume resources. We discuss a system for controlling the activities of mobile agents that uses electronic cash, a banking system, and a set of resource managers. We describe protocols for transactions between agents. We present fixed-pricing and dynamic-pricing policies for resources. We focus on and analyze the sealed-bid second-price auction as a mechanism for dynamic pricing.
- [BKR98b] Jonathan Bredin, David Kotz, and Daniela Rus. Utility driven mobile-agent scheduling. Unpublished, October 1998.
Abstract: Mobile agents are programs capable of migrating from one host machine to another. We propose that mobile agents purchase resource access rights from host machines thereby establishing a market for computational resources and giving agents a metric to evenly distribute themselves throughout the network. Market participation requires quantitative information about resource consumption to define demand and calculate utility.
 We create a formal utility model to derive user-demand functions, allowing agents to efficiently plan expenditure and deal with price fluctuations. By quantifying demand and utility, resource owners can precisely set a value for a good. We simulate our model in a mobile agent scheduling environment and show how mobile agents may use server prices to distribute themselves evenly throughout a network.
- [BKR98c] Jonathan Bredin, David Kotz, and Daniela Rus. [Utility driven mobile-agent scheduling](#). Technical Report PCS-TR98-331, Dept. of Computer Science, Dartmouth College, May 1998. Revised October 3, 1998.
Abstract: Mobile agents are programs capable of migrating from one host machine to another. We propose that mobile agents purchase resource access rights from host machines thereby establishing a market for computational resources and giving agents a metric to evenly distribute themselves throughout the network. Market participation requires quantitative information about resource consumption to define demand and calculate utility.
 We create a formal utility model to derive user-demand functions, allowing agents to efficiently plan expenditure and deal with price fluctuations. By quantifying demand and utility, resource owners can precisely set a value for a good. We simulate our model in a mobile agent scheduling environment and show how mobile agents may use server prices to distribute themselves evenly throughout a network.
- [BKR99a] Jonathan Bredin, David Kotz, and Daniela Rus. [Economic markets as a means of open mobile-agent systems](#). In *Proceedings of the Workshop “Mobile Agents in the Context of Competition and Cooperation (MAC3)” at Autonomous Agents ’99*, pages 43–49, May 1999.
Abstract: Mobile-agent systems have gained popularity in use because they ease the application design

process by giving software engineers greater flexibility. Although the value of any network is dependent on both the number of users and the number of sites participating in the network, there is little motivation for systems to donate resources to arbitrary agents. We propose to remedy the problem by imposing an economic market on mobile-agent systems where agents purchase resources from host sites and sell services to users and other agents. Host sites accumulate revenues, which are distributed to users to be used to launch more agents. We argue for the use of markets to regulate mobile-agent systems and discuss open issues in implementing market-based mobile-agent systems.

- [BKR99b] Jonathan Bredin, David Kotz, and Daniela Rus. Mobile-agent planning in a market-oriented environment. Accepted at, and withdrawn from, ASA/MA '99, August 1999.
Abstract: We propose a method for increasing incentives for sites to host arbitrary mobile agents in which mobile agents purchase their computing needs from host sites. We present a scalable market-based CPU allocation policy and an on-line algorithm that plans a mobile agent's expenditure over a multihop ordered itinerary. The algorithm chooses a set of sites at which to execute and computational priorities at each site to minimize execution time while preserving a prespecified budget constraint. We present simulation results of our algorithm to show that our allocation policy and planning algorithm scale well as more agents are added to the system.
- [BKR99c] Jonathan Bredin, David Kotz, and Daniela Rus. [Mobile-agent planning in a market-oriented environment](#). Technical Report PCS-TR99-345, Dept. of Computer Science, Dartmouth College, May 1999. Revision 1 of May 20, 1999.
Abstract: We propose a method for increasing incentives for sites to host arbitrary mobile agents in which mobile agents purchase their computing needs from host sites. We present a scalable market-based CPU allocation policy and an on-line algorithm that plans a mobile agent's expenditure over a multihop ordered itinerary. The algorithm chooses a set of sites at which to execute and computational priorities at each site to minimize execution time while preserving a prespecified budget constraint. We present simulation results of our algorithm to show that our allocation policy and planning algorithm scale well as more agents are added to the system.
- [BKR00] Jonathan Bredin, David Kotz, and Daniela Rus. [Trading risk in mobile-agent computational markets](#). In *Persented at the Sixth International Conference on Computing in Economics and Finance*, Barcelona, Spain, July 2000. No proceedings available.
Abstract: Mobile-agent systems allow user programs to autonomously relocate from one host site to another. This autonomy provides a powerful, flexible architecture on which to build distributed applications. The asynchronous, decentralized nature of mobile-agent systems makes them flexible, but also hinders their deployment. We argue that a market-based approach where agents buy computational resources from their hosts solves many problems faced by mobile-agent systems.
- In our earlier work, we propose a policy for allocating general computational priority among agents posed as a competitive game for which we derive a unique computable Nash equilibrium. Here we improve on our earlier approach by implementing resource guarantees where mobile-agent hosts issue call options on computational resources. Call options allow an agent to reserve and guarantee the cost and time necessary to complete its itinerary before the agent begins execution.
- We present an algorithm based upon the binomial options-pricing model that estimates future congestion to allow hosts to evaluate call options; methods for agents to measure the risk associated with their performance and compare their expected utility of competing in the computational spot market with utilizing resource options; and test our theory with simulations to show that option trade reduces variance in agent completion times.
- [BKR01a] Jonathan Bredin, David Kotz, and Daniela Rus. [The role of information in computational-resource allocation, for the TASK electronic commerce REF](#). Invited paper at the DARPA TASK PI meeting, May 2001.
- [BKR⁺01b] Jonathan Bredin, David Kotz, Daniela Rus, Rajiv T. Maheswaran, Çağrı Imer, and Tamer Başar. [A market-based model for resource allocation in agent sys-](#)

tems. In Franco Zambonelli, editor, *Coordination of Internet Agents Models, Technologies, and Applications*, chapter 17, pages 426–441. Springer-Verlag, 2001.

Abstract: In traditional computational systems, resource owners have no incentive to subject themselves to additional risk and congestion associated with providing service to arbitrary agents, but there are applications that benefit from open environments. We argue for the use of markets to regulate agent systems. With market mechanisms, agents have the abilities to assess the cost of their actions, behave responsibly, and coordinate their resource usage both temporally and spatially.

We discuss our market structure and mechanisms we have developed to foster secure exchange between agents and hosts. Additionally, we believe that certain agent applications encourage repeated interactions that benefit both agents and hosts, giving further reason for hosts to fairly accommodate agents. We apply our ideas to create a resource-allocation policy for mobile-agent systems, from which we derive an algorithm for a mobile agent to plan its expenditure and travel. With perfect information, the algorithm guarantees the agent’s optimal completion time.

We relax the assumptions underlying our algorithm design and simulate our planning algorithm and allocation policy to show that the policy prioritizes agents by endowment, handles bursty workloads, adapts to situations where network resources are overextended, and that delaying agents’ actions does not catastrophically affect agents’ performance.

- [BMC99] Carl J. Beckmann, Donald D. McManus, and George Cybenko. [Horizons in scientific and distributed computing](#). *IEEE Computing in Science and Engineering*, 1(1):23–30, January 1999.
Abstract: The article surveys current technologies relevant to developing scientific applications on globally distributed networks. It reviews some networking and computing technologies that are having a significant impact, real or perceived, in the commercial computing world and might be valuable for future distributed scientific computing. In many cases, an unfortunate combination of technical inbreeding and aggressive marketing has created jargon and hyperbole barriers to understanding. So, presentations of these technologies too often use terminology potentially foreign to scientific computing people. That has been our experience at least. The article’s goal is to remove some of these barriers. Developing scientific applications on globally distributed networks requires language support (Java, MPI, OpenMP), mechanisms for managing distributed computations and services (components and agents), and advanced networking technologies (IPv6, ATM).
- [BMcI⁺99] Jonathan Bredin, Rajiv T. Maheswaran, Çağrı Imer, Tamer Başar, David Kotz, and Daniela Rus. [A game-theoretic formulation of multi-agent resource allocation](#). Technical Report PCS-TR99-360, Dept. of Computer Science, Dartmouth College, October 1999.
Abstract: This paper considers resource allocation in a network with mobile agents competing for computational priority. We formulate this problem as a multi-agent game with the players being agents purchasing service from a common server. We show that there exists a computable Nash equilibrium when agents have perfect information into the future. We simulate a network of hosts and agents using our strategy to show that our resource-allocation mechanism effectively prioritizes agents according to their endowments.
- [BMcI⁺00] Jonathan Bredin, Rajiv T. Maheswaran, Çağrı Imer, Tamer Başar, David Kotz, and Daniela Rus. [A game-theoretic formulation of multi-agent resource allocation](#). In *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 349–356. ACM Press, June 2000.
Abstract: This paper considers resource allocation in a network with mobile agents competing for computational priority. We formulate this problem as a multi-agent game with the players being agents purchasing service from a common server. We show that there exists a computable Nash equilibrium when agents have perfect information into the future. From our game, we build a market-based CPU allocation policy and a strategy with which an agent may plan its expenditures for a multi-hop itinerary. We simulate a network of hosts and agents using our strategy to show that our resource-allocation mechanism effectively prioritizes agents according to their endowments and that our planning algorithm handles network delay gracefully.

- [BMcI⁺03] Jonathan Bredin, Rajiv T. Maheswaran, Çağrı Imer, Tamer Başar, David Kotz, and Daniela Rus. [Computational markets to regulate mobile-agent systems](#). *Autonomous Agents and Multi-Agent Systems*, 6(3):235–263, May 2003.

Abstract: Mobile-agent systems allow applications to distribute their resource consumption across the network. By prioritizing applications and publishing the cost of actions, it is possible for applications to achieve faster performance than in an environment where resources are evenly shared. We enforce the costs of actions through markets where user applications bid for computation from host machines.

We represent applications as collections of mobile agents and introduce a distributed mechanism for allocating general computational priority to mobile agents. We derive a bidding strategy for an agent that plans expenditures given a budget and a series of tasks to complete. We also show that a unique Nash equilibrium exists between the agents under our allocation policy. We present simulation results to show that the use of our resource-allocation mechanism and expenditure-planning algorithm results in shorter mean job completion times compared to traditional mobile-agent resource allocation. We also observe that our resource-allocation policy adapts favorably to allocate overloaded resources to higher priority agents, and that agents are able to effectively plan expenditures even when faced with network delay and job-size estimation error.

- [Bre01a] Jonathan L. Bredin. *Market-based Control of Mobile-agent Systems*. PhD thesis, Dept. of Computer Science, Dartmouth College, June 2001. Available as Dartmouth Computer Science Technical Report TR2001-408.

Abstract: Modern distributed systems scatter sensors, storage, and computation throughout the environment. Ideally these devices communicate and share resources, but there is seldom motivation for a device’s owner to yield control to another user. We establish markets for computational resources to motivate principals to share resources with arbitrary users, to enforce priority in distributed systems, to provide flexible and rational limitations on the potential of an application, and to provide a lightweight structure to balance the workload over time and between devices. As proof of concept, we implement a structure software agents can use to discover and negotiate access to networked resources. The structure separates discovery, authentication, and consumption enforcement as separate orthogonal issues to give system designers flexibility.

Mobile agents represent informational and computational flow. We develop mechanisms that distributively allocate computation among mobile agents in two settings. The first models a situation where users collectively own networked computing resources and require priority enforcement. We extend the allocation mechanism to allow resource reservation to mitigate utility volatility. The second, more general model relaxes the ownership assumption. We apply our computational market to an open setting where a principal’s chief concern is revenue maximization.

Our simulations compare the performance of market-based allocation policies to traditional policies and relate the cost of ownership and consumption separation. We observe that our markets effectively prioritize applications’ performance, can operate under uncertainty and network delay, provide metrics to balance network load, and allow measurement of market-participation risk versus reservation-based computation.

In addition to allocation problems, we investigate resource selection to optimize execution time. The problem is NP-complete if the costs and latencies are constant. Both metrics’ dependence on the chosen set complicates matters. We study how a greedy approach, a novel heuristic, and a shortest-constrained-path strategy perform in mobile-agent applications.

Market-based computational-resource allocation fertilizes applications where previously there was a dearth of motive for or means of cooperation. The rationale behind mobile-agent performance optimization is also useful for resource allocation in general distributed systems where an application has a sequence of dependent tasks or when data collection is expensive.

- [Bre01b] Jonathan L. Bredin. [Market-based control of mobile-agent systems](#). Technical Report TR2001-408, Dept. of Computer Science, Dartmouth College, June 2001. Ph.D Dissertation.

Abstract: Modern distributed systems scatter sensors, storage, and computation throughout the environment. Ideally these devices communicate and share resources, but there is seldom motivation for a device's owner to yield control to another user. We establish markets for computational resources to motivate principals to share resources with arbitrary users, to enforce priority in distributed systems, to provide flexible and rational limitations on the potential of an application, and to provide a lightweight structure to balance the workload over time and between devices. As proof of concept, we implement a structure software agents can use to discover and negotiate access to networked resources. The structure separates discovery, authentication, and consumption enforcement as separate orthogonal issues to give system designers flexibility.

Mobile agents represent informational and computational flow. We develop mechanisms that distributively allocate computation among mobile agents in two settings. The first models a situation where users collectively own networked computing resources and require priority enforcement. We extend the allocation mechanism to allow resource reservation to mitigate utility volatility. The second, more general model relaxes the ownership assumption. We apply our computational market to an open setting where a principal's chief concern is revenue maximization.

Our simulations compare the performance of market-based allocation policies to traditional policies and relate the cost of ownership and consumption separation. We observe that our markets effectively prioritize applications' performance, can operate under uncertainty and network delay, provide metrics to balance network load, and allow measurement of market-participation risk versus reservation-based computation.

In addition to allocation problems, we investigate resource selection to optimize execution time. The problem is NP-complete if the costs and latencies are constant. Both metrics' dependence on the chosen set complicates matters. We study how a greedy approach, a novel heuristic, and a shortest-constrained-path strategy perform in mobile-agent applications.

Market-based computational-resource allocation fertilizes applications where previously there was a dearth of motive for or means of cooperation. The rationale behind mobile-agent performance optimization is also useful for resource allocation in general distributed systems where an application has a sequence of dependent tasks or when data collection is expensive.

- [CC03] Valentino Crespi and George Cybenko. [Decentralized algorithms for sensor registration](#). In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 266–271, Portland, OR, July 2003. IEEE Computer Society Press.
Abstract: In this paper we investigate a problem arising in decentralized registration of sensors. The application we consider involves a heterogeneous collection of sensors—some sensors have on-board Global Positioning System (GPS) capabilities while others do not. All sensors have wireless communications capability but the wireless communication has limited effective range. Sensors can communicate only with other sensors that are within a fixed distance of each other. Sensors with GPS capability are self-registering. Sensors without GPS capability are less expensive and smaller but they must compute estimates of their location using estimates of the distances between themselves and other sensors within their radio range. GPS-less sensors may be several radio hops away from GPS-capable sensors so registration must be inferred transitively. Our approach to solving this registration problem involves minimizing a global potential or penalty function by using only local information, determined by the radio range, available to each sensor. The algorithm we derive is a special case of a more general methodology we have developed called "Emergence Engineering".
- [CG00] Ezra E. K. Cooper and Robert S. Gray. [An economic CPU-time market for D'Agents](#). Technical Report TR2000–375, Dept. of Computer Science, Dartmouth College, June 2000.
Abstract: A usable and efficient resource-management system has been created for use with D'Agents. The software dynamically negotiates a price rate for CPU time, using the competitive bids of mobile agents that offer currency in return for fast computation. The system allows mobile agents to plan their expenditures across many hosts while minimizing the time needed for their tasks. The ability to price CPU time opens the door for service owners to be compensated for the computation consumed by agents

and provides an incentive for servers to allow anonymous agents. We discuss the theoretical background which makes a CPU market system possible and the performance of the D'Agents market system.

- [Che04a] Guanling Chen. *Solar: Building A Context Fusion Network for Pervasive Computing*. PhD thesis, Dept. of Computer Science, Dartmouth College, August 2004.
Abstract: The complexity of developing context-aware pervasive-computing applications calls for distributed software infrastructures that assist applications to collect, aggregate, and disseminate contextual data. In this dissertation, we present a Context Fusion Network (CFN), called Solar, which is built with a scalable and self-organized service overlay. Solar is flexible and allows applications to select distributed data sources and compose them with customized data-fusion operators into a directed acyclic information flow graph. Such a graph represents how an application computes high-level understandings of its execution context from low-level sensory data. To manage application-specified operators on a set of overlay nodes called Planets, Solar provides several unique services such as application-level multicast with policy-driven data reduction to handle buffer overflow, context-sensitive resource discovery to handle environment dynamics, and proactive monitoring and recovery to handle common failures. Experimental results show that these services perform well on a typical DHT-based peer-to-peer routing substrate. In this dissertation, we also discuss experience, insights, and lessons learned from our quantitative analysis of the input sensors, a detailed case study of a Solar application, and development of other applications in different domains.
- [Che04b] Guanling Chen. [Solar: Building a context fusion network for pervasive computing](#). Technical Report TR2004-514, Dept. of Computer Science, Dartmouth College, August 2004.
Abstract: The complexity of developing context-aware pervasive-computing applications calls for distributed software infrastructures that assist applications to collect, aggregate, and disseminate contextual data. In this dissertation, we present a Context Fusion Network (CFN), called Solar, which is built with a scalable and self-organized service overlay. Solar is flexible and allows applications to select distributed data sources and compose them with customized data-fusion operators into a directed acyclic information flow graph. Such a graph represents how an application computes high-level understandings of its execution context from low-level sensory data. To manage application-specified operators on a set of overlay nodes called Planets, Solar provides several unique services such as application-level multicast with policy-driven data reduction to handle buffer overflow, context-sensitive resource discovery to handle environment dynamics, and proactive monitoring and recovery to handle common failures. Experimental results show that these services perform well on a typical DHT-based peer-to-peer routing substrate. In this dissertation, we also discuss experience, insights, and lessons learned from our quantitative analysis of the input sensors, a detailed case study of a Solar application, and development of other applications in different domains.
- [CJ99] George Cybenko and Guofei Jiang. [Matching conflicts: Functional validation of agents](#). In *AAAI Workshop of Agent Conflicts*, pages 14–19, Orlando, Florida, July 1999. AAAI Press.
Abstract: In most working and proposed multiagent systems, the problem of identifying and locating agents that can provide specific services is a major problem of concern. A broker or matchmaker service is often proposed as a solution. These systems use keywords drawn from application domain ontologies to specify agent service, usually framed within some sort of knowledge representation language. However, we believe that keywords and ontologies cannot be defined and interpreted precisely enough to make broking or matchmaking among agents sufficiently robust in a truly distributed, heterogeneous, multiagent computing environment. This creates matching conflicts between a client agents' requested functionality and a service agents' actual functionality. We propose a new form of interagent communication, called functional validation, specially designed to solve such matching conflicts. In this paper we introduce the functional validation concept, analyze the possible situations that can arise in validation problems and formalize the mathematical framework around which further work can be done.
- [CJ00] George Cybenko and Guofei Jiang. [The infrastructure web: a distributed system for critical infrastructure monitoring and management](#). In *Proceedings of the Symposium of Infrastructure Protection and Emergency Management*, Washington, DC, April 2000.
Abstract: National-scale critical infrastructure protection depends on many processes: intelligence

gathering, analysis, interdiction, detection, response and recovery, to name a few. These processes are typically carried out by different individuals, agencies and industry sectors. Many new threats to national infrastructure are arising from the complex couplings that exist between advanced information technologies (telecommunications and internet), physical components (utilities), human services (health, law enforcement, emergency management) and commerce (financial services, logistics). Those threats arise and evolve at a rate governed by human intelligence and innovation, on internet timeor infrastructure protection must operate on the same time scale to be effective. To achieve this, a new approach to integrating, coordinating and managing infrastructure protection must be deployed. To this end, we describe the key ingredients of an Infrastructure Web. The Infrastructure Web is a web-like architecture for decentralized monitoring and managing critical national infrastructures.

- [CJB99] George Cybenko, Guofei Jiang, and Daniel Bilar. [Machine learning applications in grid computing](#). In *Proceedings of the 37th Allerton Conference on Communication, Control and Computing*, pages 348–357, September 1999.
Abstract: The development of the World Wide Web has changed the way that we think about information. Information on the web is distributed, updates are made asynchronously and resources come and go online without centralized control. Global networking will similarly change the way we think about and perform computation. Grid computing refers to computing in a distributed networked environment in which computing and data resources are located throughout a network. Grid computing is enabled by an infrastructure that allows users to locate computing resources and data products dynamically during a computation. In order to locate resources dynamically in a grid computation, a grid application program consults a broker or matchmaker agent that uses keywords and ontologies to specify grid services. However, we believe that keywords and ontologies cannot be defined or interpreted precisely enough to make brokering and matchmaking between agents sufficiently robust in a truly distributed, heterogeneous computing environment. To this end, we introduce the concept of functional validation. Functional validation goes beyond the symbolic negotiation level of brokering and matchmaking, to the level of validating actual functional performance of grid services. In this paper, we present the functional validation problem in grid computing and apply basic machine learning theory such as PAC learning and Chernoff bounds to solve the sample size problem that arises. Furthermore, in order to reduce network traffic and speedup the validation process, we describe the use of Dartmouth D’Agents technology to implement a general mobile functional validation agent system which can be integrated into a grid computing infrastructures as a standard grid service.
- [CK01a] Guanling Chen and David Kotz. [Solar: Towards a flexible and scalable data-fusion infrastructure for ubiquitous computing](#). In *UbiTools workshop at UbiComp 2001*, October 2001.
Abstract: As we embed more computers into our daily environment, ubiquitous computing promises to make them less noticeable and to avoid information overload. We see, however, few ubiquitous applications that are able to adapt to the dynamics of user, physical, and computational context. The challenge is to allow applications flexible access to these sources, and yet scale to thousands of devices and sensors. In this paper we introduce our proposed infrastructure, Solar. In Solar, information sources produce events. Applications may subscribe to interesting sources directly, or they may instantiate and subscribe to a tree of operators that filter, transform, merge and aggregate events. Applications use a subscription language to describe the tree, based on event streams registered in a context-sensitive naming hierarchy. Solar is flexible: modular operators can be composed to produce new event streams. Solar is scalable: it distributes operators across hosts called Planets, and it re-uses common subgraphs in the operator network.
- [CK01b] Guanling Chen and David Kotz. [Supporting adaptive ubiquitous applications with the SOLAR system](#). Technical Report TR2001-397, Dept. of Computer Science, Dartmouth College, May 2001.
Abstract: As we embed more computers into our daily environment, ubiquitous computing promises to make them less noticeable and help to prevent information overload. We see, however, few ubiquitous applications that are able to adapt to the dynamics of user, physical, and computational context. We believe that there are two challenges causing this lack of ubiquitous applications: there is no flexible and scalable way to support information collection and dissemination in a ubiquitous

and mobile environment, and there is no general approach to building adaptive applications given heterogeneous contextual information. We propose a system infrastructure, Solar, to meet these challenges. Solar uses a subscription-based operator graph abstraction and allows dynamic composition of stackable operators to manage ubiquitous information sources. After developing a set of diverse adaptive applications, we expect to identify fundamental techniques for context-aware adaptation. Our expectation is that Solar’s end-to-end support for information collection, dissemination, and utilization will make it easy to build adaptive applications for a ubiquitous mobile environment with many users and devices.

- [CK02a] Guanling Chen and David Kotz. [Context aggregation and dissemination in ubiquitous computing systems](#). In *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications*, pages 105–114. IEEE Computer Society Press, June 2002.

Abstract: Many “ubiquitous computing” applications need a constant flow of information about their environment to be able to adapt to their changing context. To support these “context-aware” applications we propose a graph-based abstraction for collecting, aggregating, and disseminating context information. The abstraction models context information as events, produced by sources and flowing through a directed acyclic graph of event-processing operators and delivered to subscribing applications. Applications describe their desired event stream as a tree of operators that aggregate low-level context information published by existing sources into the high-level context information needed by the application. The operator graph is thus the dynamic combination of all applications’ subscription trees.

In this paper, we motivate and describe our graph abstraction, and discuss a variety of critical design issues. We also sketch our Solar system, an implementation that represents one point in the design space for our graph abstraction.

- [CK02b] Guanling Chen and David Kotz. [Context aggregation and dissemination in ubiquitous computing systems](#). Technical Report TR2002-420, Dept. of Computer Science, Dartmouth College, February 2002.

Abstract: Many “ubiquitous computing” applications need a constant flow of information about their environment to be able to adapt to their changing context. To support these “context-aware” applications we propose a graph-based abstraction for collecting, aggregating, and disseminating context information. The abstraction models context information as events, produced by sources and flowing through a directed acyclic graph of event-processing operators and delivered to subscribing applications. Applications describe their desired event stream as a tree of operators that aggregate low-level context information published by existing sources into the high-level context information needed by the application. The operator graph is thus the dynamic combination of all applications’ subscription trees.

In this paper, we motivate and describe our graph abstraction, and discuss a variety of critical design issues. We also sketch our Solar system, an implementation that represents one point in the design space for our graph abstraction.

- [CK02c] Guanling Chen and David Kotz. [Solar: A pervasive-computing infrastructure for context-aware mobile applications](#). Technical Report TR2002-421, Dept. of Computer Science, Dartmouth College, February 2002.

Abstract: Emerging pervasive computing technologies transform the way we live and work by embedding computation in our surrounding environment. To avoid increasing complexity, and allow the user to concentrate on her tasks, applications must automatically adapt to their changing *context*, the physical and computational environment in which they run. To support these “context-aware” applications we propose a graph-based abstraction for collecting, aggregating, and disseminating context information. The abstraction models context information as *events*, which are produced by *sources*, flow through a directed acyclic graph of event-processing *operators*, and are delivered to subscribing applications. Applications describe their desired event stream as a tree of operators that aggregate low-level context information published by existing sources into the high-level context information needed by the application. The *operator graph* is thus the dynamic combination of all applications’ subscription trees. In this paper, we motivate our graph abstraction by discussing several

applications under development, sketch the architecture of our system (“Solar”) that implements our abstraction, report some early experimental results from the prototype, and outline issues for future research.

- [CK02d] Guanling Chen and David Kotz. [Solar: An open platform for context-aware mobile applications](#). In *Proceedings of the First International Conference on Pervasive Computing (Short paper)*, pages 41–47, June 2002. In an informal companion volume of short papers. **Abstract:** Emerging pervasive computing technologies transform the way we live and work by embedding computation in our surrounding environment. To avoid increasing complexity, and allow the user to concentrate on her tasks, applications in a pervasive computing environment must automatically adapt to their changing *context*, including the user state and the physical and computational environment in which they run. Solar is a middleware platform to help these “context-aware” applications aggregate desired context from heterogeneous sources and to locate environmental services depending on the current context. By moving most of the context computation into the infrastructure, Solar allows applications to run on thin mobile clients more effectively. By providing an open framework to enable dynamic injection of context processing modules, Solar shares these modules across many applications, reducing application development cost and network traffic. By distributing these modules across network nodes and reconfiguring the distribution at runtime, Solar achieves parallelism and online load balancing.
- [CK03] Guanling Chen and David Kotz. [Context-aware resource discovery](#). In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, pages 243–252. IEEE Computer Society Press, March 2003. **Abstract:** This paper presents the “Solar” system framework that allows resources to advertise context-sensitive names and for applications to make context-sensitive name queries. The heart of our framework is a small specification language that allows composition of “context-processing operators” to calculate the desired context. Resources use the framework to register and applications use the framework to lookup context-sensitive name descriptions. The back-end system executes these operators and constantly updates the context values, adjusting advertised names and informing applications about changes. We report experimental results from a prototype, using a modified version of the Intentional Naming System (INS) as the core directory service.
- [CK04a] Guanling Chen and David Kotz. [Application-controlled loss-tolerant data dissemination](#). Technical Report TR2004-488, Dept. of Computer Science, Dartmouth College, February 2004. **Abstract:** Reactive or proactive mobile applications require continuous monitoring of their physical and computational environment to make appropriate decisions in time. These applications need to monitor data streams produced by sensors and react to changes. When mobile sensors and applications are connected by low-bandwidth wireless networks, sensor data rates may overwhelm the capacity of network links or of the applications. In traditional networks and distributed systems, flow-control and congestion-control policies either drop data or force the sender to pause. When the data sender is sensing the physical environment, however, a pause is equivalent to dropping data. Arbitrary data drops are not necessarily acceptable to the reactive mobile applications receiving sensor data. Data distribution systems must support application-specific policies that selectively drop data objects when network or application buffers overflow.
- In this paper we present a data-dissemination service, PACK, which allows applications to specify customized data-reduction policies. These policies define how to discard or summarize data flows wherever buffers overflow on the dissemination path, notably at the mobile hosts where applications often reside. The PACK service provides an overlay infrastructure to support mobile data sources and sinks, using application-specific data-reduction policies where necessary along the data path. We uniformly apply the data-stream “packing” abstraction to buffer overflow caused by network congestion, slow receivers, and the temporary disconnections caused by end-host mobility. We demonstrate the effectiveness of our approach with an application example and experimental measurements.
- [CK04b] Guanling Chen and David Kotz. [Dependency management in distributed settings](#). Technical Report TR2004-495, Dept. of Computer Science, Dartmouth College, March 2004. **Abstract:** Ubiquitous-computing environments are heterogeneous and volatile in nature. Systems that

support ubicomp applications must be self-managed, to reduce human intervention. In this paper, we present a general service that helps distributed software components to manage their dependencies. Our service proactively monitors the liveness of components and recovers them according to supplied policies. Our service also tracks the state of components, on behalf of their dependents, and may automatically select components for the dependent to use based on evaluations of customized functions. We believe that our approach is flexible and abstracts away many of the complexities encountered in ubicomp environments. In particular, we show how we applied the service to manage dependencies of context-fusion operators and present some experimental results.

- [CK04c] Guanling Chen and David Kotz. [Dependency management in distributed settings \(poster abstract\)](#). In *International Conference on Autonomic Computing (ICAC-04)*, May 2004.
Abstract: Ubiquitous-computing environments are heterogeneous and volatile in nature. Systems that support ubicomp applications must be self-managed, to reduce human intervention. In this paper, we present a general service that helps distributed software components to manage their dependencies. Our service proactively monitors the liveness of components and recovers them according to supplied policies. Our service also tracks the state of components, on behalf of their dependents, and may automatically select components for the dependent to use based on evaluations of customized functions. We believe that our approach is flexible and abstracts away many of the complexities encountered in ubicomp environments. In particular, we show how we applied the service to manage dependencies of context-fusion operators and present some experimental results.

- [CLK04] Guanling Chen, Ming Li, and David Kotz. [Design and implementation of a large-scale context fusion network](#). In *First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous)*, August 2004. Accepted for publication.
Abstract: In this paper we motivate a Context Fusion Network (CFN), an infrastructure model that allows context-aware applications to select distributed data sources and compose them with customized data-fusion operators into a directed acyclic information fusion graph. Such a graph represents how an application computes high-level understandings of its execution context from low-level sensory data. Multiple graphs by different applications inter-connect with each other to form a global graph. A key advantage of a CFN is re-usability, both at code-level and instance-level, facilitated by operator composition. We designed and implemented a distributed CFN system, Solar, which maps the logical operator graph representation onto a set of overlay hosts. In particular, Solar meets the challenges inherent to heterogeneous and volatile ubicomp environments. By abstracting most complexities into the infrastructure, we believe Solar facilitates both the development and deployment of context-aware applications. We present the operator composition model, basic services of the Solar overlay network, and programming support for the developers. We also discuss some applications built with Solar and the lessons we learned from our experience.

- [CM99] G. Cybenko and K. Moizumi. The multiple travelling agent problem. In *Proceedings of the IFORS '99, the 15th Triennial Conference of IFORS (the International Federation of Operational Research Societies)*, Beijing, PRC, August 1999.

- [CPR03] Peter Corke, Ronald Peterson, and Daniela Rus. [Networked robots: Flying robot navigation using a sensor net](#). In *Proceedings of the Eleventh International Symposium of Robotics Research (ISRR)*, Springer Tracts on Advanced Robotics (STAR). Springer-Verlag, October 2003.
Abstract: This paper introduces the application of a sensor network to navigate a flying robot. We have developed distributed algorithms and efficient geographic routing techniques to incrementally guide one or more robots to points of interest based on sensor gradient fields, or along paths defined in terms of Cartesian coordinates. These include a distributed robot-assisted localization algorithm, a distributed communication-assisted path computation algorithm for the robot and a distributed communication-assisted navigation algorithm to guide the robot. The robot itself is an integral part of the localization process which establishes the positions of sensors which are not known a priori. The sensor network is an integral part of the computation and storage of the robot's path. We use this system in a large-scale outdoor experiment with Mote sensors to guide an autonomous helicopter along a path encoded in the

network. We also describe how a human can be guided using a simple handheld device that interfaces to this same environmental infrastructure.

- [Dub04] Nikita E. Dubrovsky. [Mobile agents simulation with DaSSF](#). Technical Report TR2004-499, Dept. of Computer Science, Dartmouth College, June 2004.
Abstract: Mobile agents are programs that can migrate from machine to machine in a network of computers and have complete control over their movement. Since the performance space of mobile agents has not been characterized fully, assessing the effectiveness of using mobile agents over a traditional client/server approach currently requires implementing an agent system and running time-consuming experiments.
- This report presents a simple mobile-agent simulation that can provide quick information on the performance and scalability of a generic information retrieval (IR) mobile-agent system under different network configurations. The simulation is built using the DaSSF and DaSSFNet frameworks, resulting in high performance and great configuration flexibility. This report also implements a real D’Agents mobile-agent IR system, measuring the performance of the system. A comparison of these real-world performance results and those given by the simulation suggest that the simulation has good accuracy in predicting the scalability of a mobile-agent system. Thus this report argues that simulation provides a good way to quickly assess the performance and scalability of an IR mobile-agent system under different network configurations.
- [GCK⁺02] Robert S. Gray, George Cybenko, David Kotz, Ronald A. Peterson, and Daniela Rus. [D’Agents: Applications and performance of a mobile-agent system](#). *Software—Practice and Experience*, 32(6):543–573, May 2002.
Abstract: D’Agents is a mobile-agent system that is used primarily for information-retrieval applications. In this paper, we first examine two such applications, where mobile agents greatly simplify the task of providing efficient but application-specific access to remote information resources. Then we describe the D’Agents system, which supports multiple languages, Tcl, Java and Scheme, and strong mobility for Tcl and Java. After considering the D’Agents implementation, we present some recent performance and scalability experiments that compare D’Agent mobile agents with traditional client/server approaches. The experiments show that mobile agents often outperform client/server solutions, but also demonstrate the deep interaction between environmental and application parameters. The mobile-agent performance space as a whole is complex, and significant additional experiments are needed to characterize it. Finally, after discussing current and future experiments, we explore the differences between D’Agents and other mobile-agent systems.
- [GCKR00] Robert S. Gray, George Cybenko, David Kotz, and Daniela Rus. [Mobile agents: Motivations and state of the art](#). Technical Report TR2000-365, Dept. of Computer Science, Dartmouth College, 2000.
Abstract: A mobile agent is an executing program that can migrate, at times of its own choosing, from machine to machine in a heterogeneous network. On each machine, the agent interacts with stationary service agents and other resources to accomplish its task. In this chapter, we first make the case for mobile agents, discussing six strengths of mobile agents and the applications that benefit from these strengths. Although none of these strengths are unique to mobile agents, no competing technique shares all six. In other words, a mobile-agent system provides a single general framework in which a wide range of distributed applications can be implemented efficiently and easily. We then present a representative cross-section of current mobile-agent systems.
- [GCKR02] Robert S. Gray, George Cybenko, David Kotz, and Daniela Rus. Mobile agents: Motivations and state of the art. In Jeffrey Bradshaw, editor, *Handbook of Agent Technology*. AAAI/MIT Press, 2002. Accepted for publication. Draft available as Technical Report TR2000-365, Department of Computer Science, Dartmouth College.
Abstract: A mobile agent is an executing program that can migrate, at times of its own choosing, from machine to machine in a heterogeneous network. On each machine, the agent interacts with stationary service agents and other resources to accomplish its task. In this chapter, we first make the case for mobile agents, discussing six strengths of mobile agents and the applications that benefit from

these strengths. Although none of these strengths are unique to mobile agents, no competing technique shares all six. In other words, a mobile-agent system provides a single general framework in which a wide range of distributed applications can be implemented efficiently and easily. We then present a representative cross-section of current mobile-agent systems.

- [GGK⁺01] Arne Grimstrup, Robert Gray, David Kotz, Thomas Cowin, Greg Hill, Niranjan Suri, Daria Chacón, and Martin Hofmann. [Write once, move anywhere: Toward dynamic interoperability of mobile agent systems](#). Technical Report TR2001-411, Dept. of Computer Science, Dartmouth College, July 2001.
Abstract: Mobile agents are an increasingly popular paradigm, and in recent years there has been a proliferation of mobile-agent systems. These systems are, however, largely incompatible with each other. In particular, agents cannot migrate to a host that runs a different mobile-agent system. Prior approaches to interoperability have tried to force agents to use a common API, and so far none have succeeded. Our goal, summarized in the catch phrase “Write Once, Move Anywhere,” led to our efforts to develop mechanisms that support dynamic runtime interoperability of mobile-agent systems. This paper describes the Grid Mobile-Agent System, which allows agents to migrate to different mobile-agent systems.
- [GGK⁺02] Arne Grimstrup, Robert Gray, David Kotz, Maggie Breedy, Marco Carvalho, Thomas Cowin, Daria Chacón, Joyce Barton, Chris Garrett, and Martin Hofmann. [Toward dynamic interoperability of mobile agent systems](#). In *Proceedings of the Sixth IEEE International Conference on Mobile Agents*, volume 2535 of *Lecture Notes in Computer Science*, pages 106–120, October 2002.
Abstract: Mobile agents are an increasingly popular paradigm and in recent years there has been a proliferation of mobile-agent systems. These systems are, however, largely incompatible with each other. In particular, agents cannot migrate to a host that runs a different mobile-agent system. Prior approaches to interoperability have tried to force agents to use a common API and so far none have succeeded. This goal led to our efforts to develop mechanisms that support dynamic runtime interoperability of mobile-agent systems. This paper describes the *Grid Mobile-Agent System*, which allows agents to migrate to different mobile-agent systems.
- [GKCR98] Robert S. Gray, David Kotz, George Cybenko, and Daniela Rus. [D’Agents: Security in a multiple-language, mobile-agent system](#). In Giovanni Vigna, editor, *Mobile Agents and Security*, volume 1419 of *Lecture Notes in Computer Science*, pages 154–187. Springer-Verlag, 1998.
Abstract: Mobile-agent systems must address three security issues: protecting an individual machine, protecting a group of machines, and protecting an agent. In this chapter, we discuss these three issues in the context of D’Agents, a mobile-agent system whose agents can be written in Tcl, Java and Scheme. (D’Agents was formerly known as Agent Tcl.) First we discuss mechanisms existing in D’Agents for protecting an individual machine: (1) cryptographic authentication of the agent’s owner, (2) resource managers that make policy decisions based on the owner’s identity, and (3) secure execution environments for each language that enforce the decisions of the resource managers. Then we discuss our planned market-based approach for protecting machine groups. Finally we consider several (partial) solutions for protecting an agent from a malicious machine.
- [GKP⁺01a] Robert S. Gray, David Kotz, Ronald A. Peterson, Jr., Joyce Barton, Daria Chacón, Peter Gerken, Martin Hofmann, Jeffrey Bradshaw, Maggie Breedy, Renia Jeffers, and Niranjan Suri. [Mobile-agent versus client/server performance: Scalability in an information-retrieval task](#). In *Proceedings of the Fifth IEEE International Conference on Mobile Agents*, volume 2240 of *Lecture Notes in Computer Science*, pages 229–243, Atlanta, Georgia, December 2001. Springer-Verlag. A corrected version of this paper is available on the Dartmouth web site.
Abstract: Building applications with mobile agents often reduces the bandwidth required for the application, and improves performance. The cost is increased server workload. There are, however, few studies of the scalability of mobile-agent systems. We present scalability experiments that compare four mobile-agent platforms with a traditional client/server approach. The four mobile-agent platforms

have similar behavior, but their absolute performance varies with underlying implementation choices. Our experiments demonstrate the complex interaction between environmental, application, and system parameters.

- [GKP⁺01b] Robert S. Gray, David Kotz, Ronald A. Peterson, Jr., Peter Gerken, Martin Hofmann, Daria Chacón, Greg Hill, and Niranjan Suri. [Mobile-agent versus client/server performance: Scalability in an information-retrieval task](#). Technical Report TR2001-386, Dept. of Computer Science, Dartmouth College, January 2001. **Abstract:** Mobile agents are programs that can jump from host to host in the network, at times and to places of their own choosing. Many groups have developed mobile-agent software platforms, and several mobile-agent applications. Experiments show that mobile agents can, among other things, lead to faster applications, reduced bandwidth demands, or less dependence on a reliable network connection. There are few if any studies of the scalability of mobile-agent servers, particularly as the number of clients grows. We present some recent performance and scalability experiments that compare three mobile-agent platforms with each other and with a traditional client/server approach. The experiments show that mobile agents often outperform client/server solutions, but also demonstrate the deep interaction between environmental and application parameters. The three mobile-agent platforms have similar behavior but their absolute performance varies with underlying implementation choices.
- [Gra00] Robert S. Gray. [Soldiers, agents and wireless networks: a report on a military application](#). In *Proceedings of the Fifth International Conference on the Practical Application of Intelligent Agent and Multi Agent Technology*, pages 1–6, April 2000. **Abstract:** Ideally soldiers in the field would have portable computing devices, which, connected with a wireless network, would provide access to military databases, terrain maps, and other soldiers. In this report and the associated invited talk, the author presents some routing and mobilecode technologies that have been developed to support soldiers in the field, as well as the highly intelligent network-sensing and planning agents that will be needed to fully solve the networking problems.
- [JC02] Guofei Jiang and George Cybenko. [Query routing optimization in sensor communication networks](#). In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, volume 2, pages 1999–2004, Las Vegas, NV, December 2002. IEEE Computer Society Press. **Abstract:** Interest in large-scale sensor networks for both civilian and military applications is burgeoning. The deployment of such networks will require new approaches in resource discovery, query processing and data routing. This paper presents a framework and some analytic results for query satisfaction and data routing in networks consisting of clients, sensors and data filtering/fusion servers. In this model, multiple clients pose queries that are satisfied by processing a set of sensor data streams through a set of filters or fuselets. Fuselets are lightweight data fusion algorithms that can be deployed in a network environment. The queries can have common subexpressions (sub-queries) that should be reused by multiple clients if possible and appropriate. Moreover, effective routing of data streams from sensors to clients requires routing the streams through network nodes that can implement the required filtering/fusion operations. We formulate these problems quantitatively and propose a dynamic programming based solution using sensor-fuselet location and performance tables. The framework is preliminary in that many details and variations are abstracted or ignored. However, at the end of the paper we discuss several directions that can be explored to make these preliminary results more relevant to real scenarios.
- [JCC03] Guofei Jiang, Wayne Chung, and George Cybenko. [Semantic agent technologies for tactical sensor networks](#). In Edward M. Carapezza, editor, *Proceedings of the SPIE Conference on Unattended Ground Sensor Technologies and Applications V*, volume 5090, pages 311–320, Orlando, FL, September 2003. SPIE. **Abstract:** Recent advances in wireless communication and microelectronics have enabled the development of low-cost sensor devices leading to interest in large-scale sensor networks for military applications. Sensor networks consist of large numbers of networked sensors that can be dynamically deployed and used for tactical situational awareness. One critical challenge is how to dynamically integrate these sensor networks with information fusion processes to support real-time sensing, exploitation

and decision-making in a rich tactical environment. In this paper, we describe our work on an extensible prototype to address the challenge. The prototype and its constituent technologies provide a proof-of-concept that demonstrates several fundamental new approaches for implementing next generation battlefield information systems. Many cutting-edge technologies are used to implement this system, including semantic web, web services, peer-to-peer network and content-based routing. This prototype system is able to dynamically integrate various distributed sensors and multi-level information fusion services into new applications and run them across a distributed network to support different mission goals. Agent technology plays a role in two fundamental ways: resources are described, located and tasked using semantic descriptions based on ontologies and semantic services; tracking, fusion and decision-making logic is implemented using agent objects and semantic descriptions as well.

- [JCH03] Guofei Jiang, George Cybenko, and Jim Hendler. [Semantic depth and markup complexity](#). In *Proceedings of the IEEE Conference on System, Man and Cybernetics*, pages 2138–2143, Washington, DC, October 2003. IEEE Computer Society Press.
Abstract: In order to achieve interoperability among heterogeneous systems, markup languages such as XML and DAML are being used to describe distributed systems and data. The ability to successfully interoperate based on semantic markup depends on the ability to create, use and manage shared ontologies of concepts and their interrelationships. Specifically, communicating systems in a networked environment have to achieve a certain level of semantic agreement for them to understand and process exchanged data. A challenging question is how deep the semantic agreement has to be in order to satisfy the communication needs in an environment. Additionally, what is the markup complexity resulting from pursuing that depth of semantic agreement? This paper introduces the concept of semantic depth and markup complexity and proposes models to measure the markup complexity. Furthermore, it is shown that markup complexity can be reduced by employing hierarchical ontologies after partitioning the domain into smaller sub-domains.
- [JGWC99] Guofei Jiang, Hui-Qi Gao, Cang-Pu Wu, and George Cybenko. [Convergence analysis of a discretization procedure in Q-Learning](#). In *Proceedings of the 14th IFAC Triennial World Congress*, volume Q, pages 281–286, Beijing, PRC, July 1999. Elsevier.
- [KCG⁺00] David Kotz, George Cybenko, Robert S. Gray, Guofei Jiang, Ronald A. Peterson, Martin O. Hofmann, Daria A. Chacon, Kenneth R. Whitebread, and James Hendler. [Performance analysis of mobile agents for filtering data streams on wireless networks](#). Technical Report TR2000-377, Dept. of Computer Science, Dartmouth College, October 2000.
Abstract: Wireless networks are an ideal environment for mobile agents, since their mobility allows them to move across an unreliable link to reside on a wired host, next to or closer to the resources that they need to use. Furthermore, client-specific data transformations can be moved across the wireless link and run on a wired gateway server, reducing bandwidth demands. In this paper we examine the tradeoffs faced when deciding whether to use mobile agents in a data-filtering application where numerous wireless clients filter information from a large data stream arriving across the wired network. We develop an analytical model and use parameters from filtering experiments conducted during a U.S. Navy Fleet Battle Experiment (FBE) to explore the model’s implications.
- [KCG⁺02] David Kotz, George Cybenko, Robert S. Gray, Guofei Jiang, Ronald A. Peterson, Martin O. Hofmann, Daria A. Chacón, Kenneth R. Whitebread, and James Hendler. [Performance analysis of mobile agents for filtering data streams on wireless networks](#). *Mobile Networks and Applications*, 7(2):163–174, April 2002.
Abstract: Wireless networks are an ideal environment for mobile agents, since their mobility allows them to move across an unreliable link to reside on a wired host, next to or closer to the resources that they need to use. Furthermore, client-specific data transformations can be moved across the wireless link and run on a wired gateway server, reducing bandwidth demands. In this paper we examine the tradeoffs faced when deciding whether to use mobile agents in a data-filtering application where numerous wireless clients filter information from a large data stream arriving across the wired network. We develop an analytical model and use parameters from filtering experiments conducted during a U.S. Navy Fleet Battle Experiment (FBE) to explore the model’s implications.

- [KG99a] David Kotz and Robert S. Gray. [Mobile agents and the future of the Internet](#). *ACM Operating Systems Review*, 33(3):7–13, August 1999.
Abstract: Use of the Internet has exploded in recent years with the appearance of the World-Wide Web. In this paper, we show how current technological trends may lead to a system based substantially on mobile code, and in many cases, mobile agents. We discuss several technical and non-technical hurdles along the path to that eventuality. It seems likely that, within a few years, nearly all major Internet sites will be capable of hosting and willing to host some form of mobile code or mobile agents.
- [KG99b] David Kotz and Robert S. Gray. [Mobile code: The future of the Internet](#). In *Proceedings of the Workshop “Mobile Agents in the Context of Competition and Cooperation (MAC3)” at Autonomous Agents ’99*, pages 6–12, May 1999.
Abstract: Use of the Internet has exploded in recent years with the appearance of the World-Wide Web. In this paper, we show how current technological trends necessarily lead to a system based substantially on mobile code, and in many cases, mobile agents. We discuss several technical and non-technical hurdles along the path to that eventuality. Finally, we predict that, within five years, nearly all major Internet sites will be capable of hosting and willing to host some form of mobile agents.
- [KGR02a] David Kotz, Robert Gray, and Daniela Rus. [Future directions for mobile-agent research](#). *IEEE Distributed Systems Online*, 3(8), August 2002. Based on a conversation with Jeff Bradshaw, Colin Harrison, Guenter Karjoth, Amy Murphy, Gian Pietro Picco, M. Ranganathan, Niranjani Suri, and Christian Tschudin.
Abstract: The field of mobile agents should shift its emphasis toward mobile code, in all its forms, rather than continue focusing on mobile agents. The development of modular components will help application designers take advantage of code mobility without having to rewrite their applications to fit in monolithic, mobile agent systems.
- [KGR02b] David Kotz, Robert Gray, and Daniela Rus. [Future directions for mobile-agent research](#). Technical Report TR2002-415, Dept. of Computer Science, Dartmouth College, January 2002. Based on a conversation with Jeff Bradshaw, Colin Harrison, Guenter Karjoth, Amy Murphy, Gian Pietro Picco, M. Ranganathan, Niranjani Suri, and Christian Tschudin.
Abstract: During a discussion in September 2000 the authors examined the future of research on mobile agents and mobile code. (A mobile agent is a running program that can move from host to host in network at times and to places of its own choosing.) In this paper we summarize and reflect on that discussion. It became clear that the field should shift its emphasis toward mobile code, in all its forms, rather than to continue its narrow focus on mobile agents. Furthermore, we encourage the development of modular components, so that application designers may take advantage of code mobility without needing to rewrite their application to fit in a monolithic mobile-agent system. There are many potential applications that may productively use mobile code, but there is no “killer application” for mobile agents. Finally, we note that although security is an important and challenging problem, there are many applications and environments with security requirements well within the capability of existing mobile-code and mobile-agent frameworks.
- [KJG⁺00a] David Kotz, Guofei Jiang, Robert Gray, George Cybenko, and Ronald A. Peterson. [Performance analysis of mobile agents for filtering data streams on wireless networks](#). In *Proceedings of the Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 85–94. ACM Press, August 2000.
Abstract: Wireless networks are an ideal environment for mobile agents, because their mobility allows them to move across an unreliable link to reside on a wired host, next to or closer to the resources they need to use. Furthermore, client-specific data transformations can be moved across the wireless link, and run on a wired gateway server, with the goal of reducing bandwidth demands. In this paper we examine the tradeoffs faced when deciding whether to use mobile agents to support a data-filtering application, in which numerous wireless clients filter information from a large data stream arriving across the wired network. We develop an analytical model and use parameters from our own experiments to explore the model’s implications.

- [KJG⁺00b] David Kotz, Guofei Jiang, Robert Gray, George Cybenko, and Ronald A. Peterson. [Performance analysis of mobile agents for filtering data streams on wireless networks](#). Technical Report TR2000-366, Dept. of Computer Science, Dartmouth College, May 2000.
Abstract: Wireless networks are an ideal environment for mobile agents, because their mobility allows them to move across an unreliable link to reside on a wired host, next to or closer to the resources they need to use. Furthermore, client-specific data transformations can be moved across the wireless link, and run on a wired gateway server, with the goal of reducing bandwidth demands. In this paper we examine the tradeoffs faced when deciding whether to use mobile agents to support a data-filtering application, in which numerous wireless clients filter information from a large data stream arriving across the wired network. We develop an analytical model and use parameters from our own experiments to explore the model’s implications.
- [KSP⁺03] George Kantor, Sanjiv Singh, Ron Peterson, Daniela Rus, Aveek Das, Vijay Kumar, Guilherme Pereira, and John Spletzer. [Distributed search and rescue with robot and sensor team](#). In *Proceedings of the Fourth International Conference on Field and Service Robotics*, pages 327–332. Sage Publications, July 2003. In draft proceedings distributed only at the conference.
Abstract: We develop a network of distributed mobile sensor systems as a solution to the emergency response problem. The mobile sensors are inside a building and they form a connected ad-hoc network. We discuss cooperative localization algorithms for these nodes. The sensors collect temperature data and run a distributed algorithm to assemble a temperature gradient. The mobile nodes are controlled to navigate using this temperature gradient. We also discuss how such networks can assist human users to find an exit. We have conducted an experiment to at a facility used to train firefighters to understand the environment and to test component technology. Results from experiments at this facility as well as simulations are presented here.
- [LAR03] Qun Li, Javed Aslam, and Daniela Rus. [Distributed energy-conserving routing protocols for sensor network](#). In *Proceedings of the 37th Hawaii International Conference on System Science*, January 2003.
Abstract: In this paper, we introduce three new distributed routing algorithms in sensor networks: a distributed minimal power algorithm, a distributed max-min power algorithm, and the distributed max-min αP_{min} power-aware algorithm. The first two algorithms are used to define the third, although they are very interesting and useful in their own right for applications where the optimization criterion is the minimum power, respectively the maximum residual power. By running those algorithms, the nodes in the network reduce the communication by adding a waiting time prior to each broadcast. In this way, some of the messages that travel along sub-optimal paths are suppressed. Only the messages that travel along the best paths end up being broadcast.
- [LDR03] Qun Li, Michael De Rosa, and Daniela Rus. [Distributed algorithms for guiding navigation across a sensor network](#). In *Proceedings of the Ninth Annual International Conference on Mobile Computing and Networking*, pages 313–325, San Diego, September 2003. ACM Press.
Abstract: We develop distributed algorithms for self-organizing sensor networks that respond to directing a target through a region. The sensor network models the danger levels sensed across its area and has the ability to adapt to changes. It represents the dangerous areas as obstacles. A protocol that combines the artificial potential field of the sensors with the goal location for the moving object guides the object incrementally across the network to the goal, while maintaining the safest distance to the danger areas. We give the analysis to the protocol and report on hardware experiments using a physical sensor network consisting of Mote sensors.
- [Li04] Qun Li. *Mobility and Communication in Sensor Networks*. PhD thesis, Dept. of Computer Science, Dartmouth College, August 2004.
Abstract: This thesis considered the duality between two important issues in sensor network research: communication and mobility. We build on the infrastructure of power-aware communication and global clock synchronization and show the duality between communication and mobility can be achieved to enhance each other’s quality and efficiency. First, sensor network provides a way to augment the environment for a variety of problems, including mobility-related problems such as robot

navigation. By shifting some burden of the problem to the sensor network augmented environment, the new information embedded in the environment that can be obtained by a user on spot and in real time can help to solve problems more efficiently and realistically. Second, mobility can serve to achieve communication since mobility is very common in everyday life. It is useful to use the controlled mobility of the specialized communication nodes and free natural mobility to guarantee communication, reduce power consumption, and increase network capacity.

To build an infrastructure for sensor network, we focused on two problems: power-aware communication and clock synchronization. We gave several communication protocols to conserve the energy in sensor network communication, both on the scale of the whole network and on a single node. We showed that by carefully designed routing protocol and fine-tuned sleep/wakeup node schedule, much energy can be conserved. We also designed several protocols for global clock synchronization. The most interesting one is diffusion-based clock synchronization, which is a fault-tolerant and localized protocol.

The duality between communication and mobility was shown as follows. First, we showed that communication can be achieved by controlled mobility and natural mobility. We used active trajectory change to obtain guaranteed message delivery. Then we demonstrated that natural mobility can be used to help communication to conserve energy and overcome disconnection. Second, we showed in navigation application that communication can assist mobility. We gave communication protocols to support user guidance in a sensor network, refined the protocols by considering reducing network searching space, and explored a mobility coordination problem: task assignment in robotic network applications.

- [LPDR03] Qun Li, Ron Peterson, Michael DeRosa, and Daniela Rus. [Re-active behavior in self-reconfiguring sensor network](#). *ACM Mobile Computing and Communications Review*, 7(1):56–68, January 2003.
Abstract: A self-reconfiguring sensor network consists of many sensors that have the ability to self-configure by turning themselves on and off. This kind of self-reconfiguration results in power savings and extends the lifetime of the network. In this paper we present a formulation for the problem of adapting a sensor network to the environment and the task. We develop distributed algorithms for self-reconfiguring sensor networks that respond to tracking a target and directing a target through a region.
- [LPRR02] Qun Li, Ron Peterson, Michael De Rosa, and Daniela Rus. [Reactive behavior in self-reconfiguring sensor networks](#). In *Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking*. ACM Press, September 2002. Poster abstract; later published in MC2R.
Abstract: A self-reconfiguring sensor network consists of many sensors that have the ability to self-configure by turning themselves on and off. This kind of self-reconfiguration results in power savings and extends the lifetime of the network. In this paper we present a formulation for the problem of adapting a sensor network to the environment and the task. We develop distributed algorithms for self-reconfiguring sensor networks that respond to tracking a target and directing a target through a region.
- [LR02] Qun Li and Daniela Rus. [Message relay in disconnected ad-hoc networks](#). In *IEEE MASCOTS Workshop on Mobility and Wireless Access*, October 2002.
Abstract: An ad-hoc network is formed by a group of mobile hosts upon a wireless network interface. Previous research in communication in ad-hoc networks has concentrated on routing algorithms which are designed for fully connected networks. The traditional approach to communication in a disconnected ad-hoc network is to let the mobile computer wait for network reconnection passively. This method may lead to unacceptable transmission delays. We propose an approach that guarantees message transmission in minimal time. In this approach, mobile hosts actively modify their trajectories to transmit messages. We develop algorithms that minimize the trajectory modifications under two different assumptions: (a) the movements of all the nodes in the system are known and (b) the movements of the hosts in the system are not known.

- [LR03] Qun Li and Daniela Rus. [Communication in disconnected ad-hoc networks using message relay](#). *Journal of Parallel and Distributed Computing*, 63(1):75–86, January 2003.
Abstract: An ad-hoc network is formed by a group of mobile hosts upon a wireless network interface. Previous research in communication in ad-hoc networks has concentrated on routing algorithms which are designed for fully connected networks. The traditional approach to communication in a disconnected ad-hoc network is to let the mobile computer wait for network reconnection passively. This method may lead to unacceptable transmission delays. We propose an approach that guarantees message transmission in minimal time. In this approach, mobile hosts actively modify their trajectories to transmit messages. We develop algorithms that minimize the trajectory modifications under two different assumptions: (a) the movements of all the nodes in the system are known and (b) the movements of the hosts in the system are not known.
- [LRR02] Qun Li, Michael De Rosa, and Daniela Rus. [Distributed algorithms for guiding navigation across a sensor net](#). Technical Report TR2002-435, Dept. of Computer Science, Dartmouth College, October 2002.
Abstract: We develop distributed algorithms for self-reconfiguring sensor networks that respond to directing a target through a region. The sensor network models the danger levels sensed across its area and has the ability to adapt to changes. It represents the dangerous areas as obstacles. A protocol that combines the artificial potential field of the sensors with the goal location for the moving object guides the object incrementally across the network to the goal, while maintaining the safest distance to the danger areas. We report on hardware experiments using a physical sensor network consisting of Mote sensors.
- [MC01] Katsuhiro Moizumi and George Cybenko. [The traveling agent problem](#). *Mathematics of Control, Signals and Systems*, 14(3):213–232, 2001.
Abstract: This paper considers a sequencing problem which arises naturally in the scheduling of software agents. We are given n sites at which a certain task might be successfully performed. The probability of success is p_i at the i th site and these probabilities are independent. Visiting site i and trying the task there requires time (or some other cost metric) t_i whether successful or not. Latencies between sites i and j are l_{ij} , that is, the travel time between those two sites. Should the task be successfully completed at a site then any remaining sites do not need to be visited. The *Travelling Agent Problem* is to find the sequence which minimizes the expected time to complete the task. The general formulation of this problem is NP-Complete. However, if the latencies are constant we show that the problem can be solved in polynomial time by sorting the ratios p_i/t_i according to decreasing value and visiting the sites in that order. This result then leads to an efficient algorithm when groups of sites form subnets in which latencies within a subnet are constant but can vary across subnets. We also study the case when there are deadlines for solving the problem in which case the goal is to maximize probability of success subject to satisfying the deadlines. Applications to mobile and intelligent agents are described.
- [MK02] Kazuhiro Minami and David Kotz. [Controlling access to pervasive information in the “Solar” system](#). Technical Report TR2002-422, Dept. of Computer Science, Dartmouth College, February 2002.
Abstract: Pervasive-computing infrastructures necessarily collect a lot of context information to disseminate to their context-aware applications. Due to the personal or proprietary nature of much of this context information, however, the infrastructure must limit access to context information to authorized persons. In this paper we propose a new access-control mechanism for event-based context-distribution infrastructures. The core of our approach is based on a conservative information-flow model of access control, but users may express discretionary relaxation of the resulting access-control list (ACL) by specifying relaxation functions. This combination of automatic ACL derivation and user-specified ACL relaxation allows access control to be determined and enforced in a decentralized, distributed system with no central administrator or central policy maker. It also allows users to express their personal balance between functionality and privacy. Finally, our infrastructure allows access-control policies to depend on context-sensitive roles, allowing great flexibility.

We describe our approach in terms of a specific context-dissemination framework, the Solar system, although the same principles would apply to systems with similar properties.

- [Moi98] Katsuhiro Moizumi. *The mobile-agent planning problem*. PhD thesis, Thayer School of Engineering, Dartmouth College, 1998.
Abstract: Mobile agents have received much attention recently as a way to efficiently access distributed resources in a low bandwidth network. Planning allows mobile agents to make the best use of the available resources. This thesis studies several planning problems that arise in mobile agent information retrieval and data-mining applications. The general description of the planning problems is as follows: We are given sites at which a certain task might be successfully performed. Each site has an independent probability of success associated with it. Visiting a site and trying the task there requires time (or some other cost matrix) regardless of whether the task is completed successfully or not. Latencies between sites, that is, the travel time between those two sites also have to be taken into account. If the task is successfully completed at a site then the remaining sites need not be visited. The planning problems involve finding the best sequence of sites to be visited, which minimizes the expected time to complete the task. We name the problems *Traveling Agent Problems* due to their analogy with the Traveling Salesman Problem. This Traveling Agent Problem is *NP*-complete in the general formulation. However, in this thesis a polynomial-time algorithm has been successfully developed to solve the problem by adding a realistic assumption to it. The assumption enforces the fact that the network consists of subnetworks where latencies between machines in the same subnetwork are constant while latencies between machines located in different subnetworks vary. Different versions of the Traveling Agent Problem are considered: (1) single agent problems, (2) multiple agent problems (multiple agents cooperate to complete the same task) and (3) deadline problems (single or multiple agents need to complete a task without violating a deadline constraint at each location in the network). Polynomial and pseudo-polynomial algorithms for these problems have been developed in this thesis. In addition to the theory and algorithm development for the various Traveling Agent Problems, a planning system that uses these algorithms was implemented. Descriptions of the mobile agent planning system with its supporting components such as network sensing system, directory service system, and clustering system, are also given in this thesis.
- [OC99] C. Okino and G. Cybenko. *Modeling and analysis of active messages in volatile networks*. In *Proceedings of the 37th Allerton Conference on Communications, Control and Computing*, Monticello, IL, 1999.
Abstract: Interest in and development of mobile agent software systems has burgeoned in the past five years. Code mobility has many attractive attributes for performance and dynamic deployment of new distributed computing and information management applications. An active message is a datagram encapsulated as a mobile agent. The agent is persistent in the network, moving from node to node under its own internal routing logic and control at the application layer.

Active messages are particularly attractive in networks that have very unreliable links such as wireless networks in which the nodes are mobile. Such networks experience frequent link failures and other changes in topology. Active messages allow data to propagate between nodes that may never have viable TCP/IP type connections.

In spite of the growing implementation interest in mobile agents and active messaging, there are essentially no analytic models or results dealing with their performance. This paper presents a simple model for active messages in a network with frequent link failures. Using this model, we develop expressions for the expected delivery time of an active message along one path as well as expected delivery time for duplicated messages traversing disjoint paths between source and destination nodes.
- [PR02] Ronald Peterson and Daniela Rus. *Interacting with a sensor network*. In *Proceedings of the Australian Conference for Robotics and Automation*, pages 105–110. Australian Robotics & Automation Association, November 2002.
Abstract: We develop distributed algorithms for sensor networks that respond by directing a target (robot or human) through a region. The sensor network models the event levels sensed across a geographical area, adapts to changes, and guides a moving object incrementally across the network.

We describe a device we call a Flashlight for interacting with the sensor field. This interaction includes collecting navigation information from the sensors in the local neighborhood, activating and deactivating specified areas of the sensor network, and detecting events in the sensor network. We report on hardware experiments using a physical sensor network consisting of Mote sensors.

- [RSX01] Daniela Rus, Clifford Stein, and Rong Xie. [Scheduling multi-task multi-agent systems](#). In *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 159–160. ACM Press, 2001. Poster abstract.
Abstract: We present a centralized and a distributed algorithm for scheduling multi-task agents in a distributed system. Each agent consists of multiple tasks that can be executed on multiple machines which correspond to resources. The machines in the system have different speeds. There are different communication delays and data transfer delays. We optimize the overall completion time. Our centralized algorithm has an upper bound on the overall completion time and is used as a module in the distributed algorithm. We present extensive simulation results.
- [Sun00] Shankar Sundaram. [Mission-flow constructor: A workflow management system using mobile agents](#). Master’s thesis, Thayer School of Engineering, Dartmouth College, Hanover, New Hampshire, May 2000.
Abstract: Developing code for the execution of a distributed, dynamic workflow requires significant effort and hence it becomes necessary to build tools that enable the creation and execution of such workflows. Compelling arguments have been made for the implementation of workflow management systems using mobile agents. Mobile agents are autonomous pieces of code that can migrate under their own control from one machine to another within a heterogeneous network. Mission-flow Constructor (MfC) is a workflow management system built on the D’Agents mobile agent system. Like its predecessor Mobile Agent Construction Environment (MACE), MfC uses the concept of visual languages and further abstracts the process of building a workflow. Agents generated by MfC are small and migrate only once. These agents hence make more optimal use of network resources than those generated by MACE. MfC generated agents also use improved communication means and incorporate some basic fault tolerance mechanisms. A set of primitive constructs that encapsulate commonly used topologies has been defined to make easier the process of workflow definition. A workflow specified using the GUI and associated annotation process is compiled to a set of D’Agents agents by making use of the visual depiction and the code fragments that define the individual modules. MfC then launches these agents to execute the various tasks associated with the workflow specified by the user.
- [Whi02a] A. Abram White. [Performance and interoperability in Solar](#). Technical Report TR2002-427, Dept. of Computer Science, Dartmouth College, May 2002.
Abstract: Ubiquitous computing promises to integrate computers into our physical environment, surrounding us with applications that are able to adapt to our dynamics. Solar is a software infrastructure designed to deliver contextual information to these applications. To serve the large number and wide variety of context-aware devices envisioned by ubiquitous computing, Solar must exhibit both high performance and the ability to interoperate with many computing platforms. We created a testing framework to measure the performance of distributed systems such as Solar, as well as a pluggable data-transfer mechanism to support the dissemination of information to heterogeneous applications. This paper explores the testing framework developed, analyzes its findings concerning the performance of the current Solar prototype, presents several optimizations to Solar and their effects, and finally discusses the design of the pluggable data-transfer mechanism.
- [Whi02b] A. Abram White. [XSLT and XQuery as operator languages](#). Technical Report TR2002-429, Dept. of Computer Science, Dartmouth College, May 2002.
Abstract: Ubiquitous computing promises to integrate computers into our physical environment, surrounding us with applications that are able to adapt to our dynamics. Solar is a software infrastructure designed to deliver contextual information to these applications. Solar represents context data as events, and uses small programs called operators to filter, merge, aggregate, or transform event streams. This paper explores the possibility of using XSLT and XQuery to build language-neutral Solar operators.

- [XRS01] Rong Xie, Daniela Rus, and Cliff Stein. [Scheduling multi-task agents](#). In *Proceedings of the Fifth IEEE International Conference on Mobile Agents*, volume 2240 of *Lecture Notes in Computer Science*, pages 260–276, Atlanta, Georgia, December 2001. Springer-Verlag.
- Abstract:** We present a centralized and a distributed algorithm for scheduling multi-task agents in a distributed system with the objective of minimizing the overall application completion time. Each agent consists of multiple tasks that can be executed on multiple machines which correspond to resources. The machine speeds and link transfer rates are heterogeneous. Our centralized algorithm has an upper bound on the overall completion time and is used as a module in the distributed algorithm. Extensive simulations show promising results of the algorithms, especially for scheduling communication-intensive multi-task agents.